УНИВЕРЗИТЕТ У ПРИШТИНИ
СА ПРИВРЕМЕНИМ СЕДИШТЕМ У
КОСОВСКОЈ МИТРОВИЦИ

ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА

Милош Н. Илић

ПРЕДИКЦИЈА ВРЕМЕНА ХЕМИЈСКИХ ТРЕТМАНА У ПОЉОПРИВРЕДНОЈ ПРОИЗВОДЊИ ЗАСНОВАНА НА DATA MINING ТЕХНИЦИ КОРИШЋЕЊЕМ БЕЖИЧНИХ КОМУНИКАЦИОНИХ СИСТЕМА

Докторска дисертација

Косовска Митровица, 2019.

UNIVERSITY OF PRISTINA
TEMPORARLY SETTLED IN
KOSOVSKA MITROVICA

FACULTY OF TECHNICAL SCIENCE

Miloš N. Ilić

TIME OF CHEMICAL TREATMENTS PREDICTION IN AGRICULTURAL PRODUCTION BASED ON DATA MINING TECHNIQUES USING WIRELESS COMMUNICATION SYSTEMS

Doctoral Dissertation

Kosovska Mitrovica, 2019.

Ментор: _____

проф. др Петар Спалевић, редовни професор

Универзитет у Приштини са привременим седиштем у Косовској Митровици

Факултет техничких наука

Чланови комисије:

_____

проф. др Синиша Илић, редовни професор

Универзитет у Приштини са привременим седиштем у Косовској Митровици

Факултет техничких наука

_____

проф. др Бошко Николић, редовни професор

Универзитет у Београду

Електротехнички факултет

Датум одбране:_____

ПРЕДИКЦИЈА ВРЕМЕНА ХЕМИЈСКИХ ТРЕТМАНА У ПОЉОПРИВРЕДНОЈ ПРОИЗВОДЊИ ЗАСНОВАНА НА DATA MINING ТЕХНИЦИ КОРИШЋЕЊЕМ БЕЖИЧНИХ КОМУНИКАЦИОНИХ СИСТЕМА

Сажетак

Успешно одређивање временског периода у коме су остварени услови за појаву болести и временског тренутка у коме је потребно обавити хемијске третмане представља комплексан проблем због сложености креирања предикционих модела чији је задатак дефинисање веза између појаве болести и тренутних метеоролошких услова. Тачност предикције времена хемијских третмана директно утиче на економичност пољопривредне производње и количине потребних пестицида, те доприноси заштити животне средине, као и здравијим пољопривредним производима. Спроведеним истраживањем је креирано софтверско решење, базирано на примени data mining техника и бежичних комуникационих система, чији је задатак прикупљање метеоролошких и просторно-временских параметара, на основу којих се врши предикција остварености услова за појаву биљних болести, а самим тим и предикција времена хемијских третмана. Развијено решење је затвореног типа, односно у оквиру истог се врши прикупљање потребних података, обрада података у циљу креирања предикционих модела и примена тих модела за предикцију времена хемијских третмана. Како би се утврдила тачност предикције, извршено је тестирање добијених предикционих модела за креиране скупове података. Закључено је да тачност предикције износи 93,71%, што оправдава коришћење система заснованог на data mining техникама и бежичним комуникационим системима за предикцију времена хемијских третмана.

Кључне речи: data mining, GSM/GPRS, J48, дискриминациона анализа, PCA

Научна област: Електротехничко и рачунарско инжењерство

Ужа научна област: Интердисциплинарна примена Data mining технике и бежичних телекомуникација

УДК број: 004.42/.62+621.39

TIME OF CHEMICAL TREATMENTS PREDICTION IN AGRICULTURAL PRODUCTION BASED ON DATA MINING TECHNIQUES USING WIRELESS COMMUNICATION SYSTEMS

Abstract

Successfully determining the time period in which the conditions for the occurrence of the disease and the time in which chemical treatments are to be made is a complex problem due to the complexity of the construction of prediction models that define the relationship between the onset of the disease and the current meteorological conditions. The accuracy of the prediction of the exact time of chemical treatments directly affects the economy of agricultural production and the amount of necessary pesticides, therefore contributes to environmental protection as well as to healthier agricultural products. Based on the conducted research, data mining software solution based on wireless communication systems is created so as to collect meteorological and spatio-temporal parameters needed for prediction of possible outbreak of diseases and the time of chemical treatments. The developed solution supports entire analysis pipeline: the collection of the necessary data, processing of data for the purpose of creating the prediction models, and the application of these models for the prediction of the time of chemical treatments. In order to determine the accuracy of the prediction, testing of the obtained prediction models for created data sets was performed. It was concluded that the prediction precision is 93.71%, which justifies the use of a system based on data mining techniques and wireless communication systems for predicting the time of chemical treatments.

Keywords: data mining, GSM/GPRS, J48, discrimination analysis, PCA.

Scientific field: Electrical and computer engineering

Scientific subfield: Interdisciplinary application of Data mining techniques and wireless telecommunications

UDC number: 004.42/.62+621.39

Садржај

1. Увод

Развој информационо комуникационих технологија, како у домену обраде великих скупова података тако и домену сензорних технологија стављених у спрегу са бежичним телекомуникационим системима, све више доприноси креирању система и софтверских решења применљивих у свакодневном животу и раду људи. Једна од новијих примена информационо комуникационих технологија јесте у области пољопривреде и производње хране. Примена ресурса информационо комуникационих технологија у пољопривредној производњи доживљава велику експанзију у последњих неколико година. Две области информационо комуникационих технологија које се посебно издвајају јесу сензорна технологија и data mining. Примена сензорних технологија, као и data mining техника, отварају један потпуно нови поглед и приступ пољопривредној производњи. Улога сензорне технологије се огледа у употреби сензора за праћење великог броја параметара директно на производним површинама. Параметри који се прате оваквим приступом крећу се од домена метеоролошких и просторно-временских параметара, преко анализе количине хранљивих материја у земљишту, па све до физиолошког стања биљака и покретања система наводњавања, прихране и заштите. Како би се обезбедила веза између сензора лоцираних на производним површинама и крајњег корисника прикупљених података, креира се комуникациони канал заснован на коришћењу бежичних телекомуникационих мрежа. Оваквим приступом је омогућено слање пакета података који садрже измерене вредности посматраних параметара и њихово чување у оквиру креираних скупова података. Креирање и коришћење великих скупова података у којима се налазе подаци од потенцијалног значаја за пољопривредну производњу, омогућавају адекватну анализу прикупљених података применом одговарајућих data mining алгоритама. Технике доступне у оквиру data mining области омогућавају обраду великих скупова података различите садржине. У домену пољопривредне производње data mining технике се, на пример, могу применити у процесу доношења одлука о времену хемијских третмана, времену у коме је потребно покренути или прекинути заливање, извршити прихрану, као и у процесу процене здравственог стања биљака, процене

могућег очекиваног приноса итд. Преглед досадашњих истраживања примене сензорних мрежа и data mining техника у оквиру процеса прикупљања података и доношења одлука дат је у другом поглављу овог рада.

Истраживање спроведено у оквиру ове дисертације имало је за циљ сагледавање могућности примене data mining техника и бежичних комуникационих система у процесу предикције времена хемијских третмана, у склопу заштите гајених биљака на производним пољопривредним површинама. Спроведено истраживање је обухватило креирање адекватних метеоролошких и просторно-временских скупова података, као и креирање и тестирање предикционих модела. Као потврда тачности креираних предикционих модела спроведена је евалуација ових модела низом већ познатих евалуационих метода. Такође, проценат тачности предикционих модела је одређен поређењем услова за појаву болести дефинисаних у оквиру креираних предикционих правила са условима дефинисаним у оквиру референтне литературе која обрађује ову област. Крајњи продукт израђене дисертације, поред научног доприноса и потврде о успешној употреби информационо комуникационих технологија за потребе предикције времена хемијских третмана, огледа се у креираном софтверском решењу доступном крајњим корисницима.

Докторска дисертација је подељена у једанаест поглавља. Поред увода који представља прво поглавље, дисертација садржи још десет тематских целина, од којих последње две тематске целине представљају закључак и преглед литературе.

Друго поглавље представља преглед досадашњих истраживања примене data mining техника и бежичних комуникационих система у пољопривредној производњи. Ово поглавље се у основи може поделити на два дела. Како је истраживање, спроведено у оквиру докторске дисерације, обухватило креирање модела метеоролошке станице намењене употреби у пољопривреди и креирање софтверског система за предикцију времена хемијских третмана, прегледом досадашњих истраживања су обухваћена значајна истраживања из домена ове две области. Најпре су описана истраживања у домену креирања сензорних мрежа за потребе праћења метеоролошких и просторно-временских параметара на пољопривредним производним површинама. У оквиру овог дела су сагледана и

истраживања у оквиру којих су измерене вредности метеоролошких и просторно-временских параметара прослеђиване до базног рачунара коришћењем бежичних комуникационих система. Другу групу истраживања представљају истраживања базирана на примени data mining техника у циљу обраде великих скупова података и креирању предикционих модела за потребе доношења одлука у пољопривредној производњи.

Треће поглавље представља методе истраживања и организационо је подељено у два дела. У оквиру овог поглавља је најпре описано место и улога data mining техника у обради великих скупова података. Након описа data mining процеса, извршен је опис метода које се користе како би се трансформисали велики скупови података. Поред метода за трансформацију великих скупова података, описане су и методе за обраду недостајућих вредности, као и методе за детекцију и отклањање *outlier*-а. Наведеним методама се успешно могу обрадити велики скупови сирових података са циљем смањења дименизионалности скупова података и отклањања недостајућих и *outlier* вредности података. Други део овог поглавља се односи на опис метода и алгоритама који су у употреби у процесу креирања и обуке предикционих модела. У овиру овог дела стављен је акценат на три групе метода: математичке и статистичке методе, класификационе data mining технике и кластеризационе data mining технике. Свака од група обухвата већи број техника и алгоритама које се могу применити на решавање проблема анализе великих скупова података, као и креирање, обуку и коришћење предикционих модела. Свака од метода се успешно примењује на решавање одређене групе проблема, па се самим тим и тачност предикционих резултата разликује у зависности од примењене методе. Такође, у оквиру сваке од група су дефинисане најадекватније методе за евалуацију успешности наведених алгоритама.

Четврто поглавље описује процес прикупљања метеоролошких и просторно-временских параметара. Ово поглавље почиње описом креираног модела метеоролошке станице намењене употреби у пољопривреди. Креирани модел метеоролошке станице је заснован на *Raspberry Pi* рачунару и скупу сензорних компоненти за прикупљање потребних метеоролошких и просторно-временских података. С тим у вези, у оквиру поглавља је дато поређење већег броја сензора за

сваки од параметара од интереса. Поређење перформанси сензора у свакој од група је извршено како би се одабрао најадекватнији сензор у погледу прецизности, напајања и утрошка енергије, као и цене самог сензора. На овакав начин се одабрани сензори могу сматрати најадекватнијим у погледу односа цена-перформансе. Такође, у оквиру креираног модела метеоролошке станице дефинисано је и напајање метеоролошке станице применом фотонапонског панела, чиме се обезбеђује мобилност овако креиране метеоролошке станице.

Пето поглавље описује процес преноса добијених вредности метеоролошких и просторно-временских параметара од метеоролошке до базне станице коришћењем бежичних комуникационих система. Први од описана два начина за пакетни пренос података, заснован је на коришћењу радио фреквенције, док је други заснован на коришћењу *GSM/GPRS* мреже. За сваки од поменута два начина преноса података су дефинисане предности и недостаци употребе, као и могући хардверски уређаји који се могу успешно повезати са *Raspberry Pi* рачунарем на предајној страни, док се на пријемној страни могу повезати са рачунаром базне станице. У циљу одабира најбољег решења, извршено је поређење перформанси хардверских уређаја обеју група.

Шесто поглавље описује креирани скуп метеоролошких и просторно-временских података. У оквиру овог поглавља је описан поступак спроведене анализе овако креираног скупа података, као и процес детекције и отклањања *outlier*-а применом неких од метода, описаних у трећем поглављу. На основу креираног скупа података, извршено је крирање два засебна скупа података који се касније користе у процесу обуке и евалуације предикционих модела.

Седмо поглавље описује креирани прототип софтверског решења за предикцију времена хемијских третмана. Прототип креираног софтверског решења је обухватио имплементацију математичких и статистичких метода, као и класификационих data mining техника. Применом различитих техника из поменуте две групе, креиран је већи број предикционих модела. У оквиру прототипа апликације, извршена је и имплементација метода за евалуацију сваког од појединачних предикционих модела, са циљем одређивања предикционог

модела који који се одликује највећом стопом тачности предикције остварености услова за појаву болести, а самим тим и предикције времена хемијских третмана.

Осмо поглавље описује функционалности и начин рада креираног софтверског система за предикцију времена хемијских третмана, под називом *GreenLife* софтверски систем. Креирани софтверски систем обухвата све функционалности од процеса прикупљања података, преко процеса креирања потребних скупова података, до процеса креирања предикционих модела и употребе истих у реалим ситуацијама. Овако креирани софтверски систем намењен је употреби од стране крајњих корисника који могу бити, како пољопривредни произвођачи тако и запослени у различитим организацијама које се баве пољопривредном производњом. Опис *GreenLife* софтверског система је дат кроз опис случајева коришћења овог система. Такође, опис система је дат кроз дефинисање архитектуре софтверског система. На овакав начин су систематски, кроз архитектурне нивое, описани алати и методи примењени у процесу имплементације овог система.

Девето поглавље представља постигнуте резултате истраживања и дискусију постигнутих резултата. У оквиру овог поглавља је дата дискусија, како утицаја правилно креираних скупова података на процес креирања и употребе предикционих модела тако и метода за унапређење предикционих модела. Такође, у оквиру поглавља је извршено поређење добијених резултата истраживања са познатим чињеницама из референтне литературе. На овакав начин је још једном потврђена тачност креираног предикционог модела и полазне хипотезе да се применом data mining метода може успешно извршити предикција времена хемијских третмана у пољопривредној производњи.

Десето поглавље представља закључак. Садржина овог поглавља се односи на сумирање целокупног истраживања кроз објашњење испуњености првобитно задатих циљева и дефинисање идеја за даљи рад и унапређење креираног софтверског система.

2. Предмет и циљеви истраживања

Убрзане климатске промене последњих година довеле су до стварања повољних метеоролошких услова за развој различитих биљних болести и појаву повећаног броја инсекатских врста које нападају гајене биљке. Све ово има за последицу смањење глобалне производње хране, што доводи до повећања проблема недостатка хране у свету. Истраживања спроведена 2002. године су показала да је у том периоду било око 840 милиона неухрањених људи на свету. Од тога, 799 милиона људи је живело у земљама у развоју, а међу њима је било око 153 милиона деце млађих од пет година. Такође се процењује да сваке године од глади умре око 6 милиона деце. Климатске промене, убрзани развој биљних болести као и смањена производња пољопривредних производа говоре у прилог чињеници да је потребно веће ангажовање на пољу производње здраве хране. Са становишта агронома и пољопривредних произвођача главни задатак у процесу производње пољопривредних производа јесте како адекватно заштитити гајене биљке на производним површинама. Заштита биљних култура на производним површинама у највећем броју случајева заснива се на хемијским третманима. С тим у вези питање које се поставља испред агронома а самим тим и пољопривредних произвођача јесте када је адекватан моменат за правилну и успешну хемијску заштиту? Ова врста проблема захтева посебну пазњу. Повећање броја организама који имају негативан утицај на гајене биљке доводи до већег броја хемијских третмана којима се површине под гајеним биљкама третирају. Велики број хемијских третмана посебно третмана у погрешном тренутку као и употреба велике количине хемијских средстава могу довести до загађења земљишта, подземних вода као и фитотоксичности гајених биљака на прозводној површини. Све ово као крајњу последицу има утицај хемијских једињења на здравље и живот људи. Остаци пестицида у људском организму могу озбиљно угрозити здравље људи. У производњи воћа као једној од грана пољопривреде у којој је химијска заштитита са једне стране неизбежна а са друге стране се велики проценат воћа конзумира у свежем стању без термичке обраде, наука и пољопривредни произвођачи морају пронаћи баланс између успешне заштите и крајњег здравог плода. Кључно решење за овај проблем састојало би се у смањењу броја превентивних третирања.

Ово практично значи да произвођачи врше хемијска третирања без прецизне информације о томе да ли су испуњени услови да до појаве болести уопште дође. Оваква третирања се обављају из бојазни да не дође до инфекције, јер су за поједине патогене/штеточине симптоми видљиви тек након инфекције, када је за заштиту већ касно. Овај проблем се може избећи коришћењем хемијских третирања у тачно одређено време, када адекватна количина препарата може довести до успешних резултата заштите од патогена/штеточина. Проблем који се јавља јесте како предвидети када је право време за заштиту, као и који услови треба да буду задовољени да би до инфекције дошло. Процес предикције се додатно усложњава чињеницом да различити патогени/штеточине нападају различите воћне врсте под различитим условима. Ово практично значи да ће патоген/штеточина инфицирати биљку само под одређеним условима. Услови под којима долази до инфекције дефинисани су у одређеним границама и може се сматрати да су унапред познати. За неке од услова параметри под којима долази до инфекције потврђени су како експериментима на терену тако и у лабораторијским условима [1]–[4].

Најважнији услови који морају бити задовољени како би дошло до инфекције јесу одговарајући метеоролошки услови и присуство активне споре патогена. Група метеоролошких параметара од којих зависи оствареност услова за инфекцију може обухватити температуру ваздуха, температуру земљишта, количину падавина, влажност ваздуха, ваздушни притисак, влажност листа биљке, смер и брзину ветра. Уколико су остваварени потребни и довољни метеоролошки услови за развој патогена/штеточина спора која се налази у активном стању моћи ће да инфицира гајену биљку. Различите биљне болести развијају се под различитим метеоролошким условима, што практично значи да под истим метеоролошким условима може доћи до развоја једних, а не мора доћи до развоја других биљних болести. Са друге стране уколико у ваздуху или на самој воћној врсти нема присуства активне споре до инфекције неће доћи без обзира што су сви метеоролошки услови испуњени. Присуство оваквих спора у ваздуху у многоме зависи од географског положаја на коме се конкретне воћне врсте гаје. На различитим надморским висинама као и на различитим експозицијама терена неће бити исти услови за могућу инфекцију.

Предмет истраживања у склопу овог рада управо због свих наведених проблема пољопривредне производње усмерен је ка проналажењу решења за благовремену предикцију времена хемијских третмана у пољопривредној производњи. Предикција времена хемијских третмана у склопу истраживања заснована је на примени data mining техника и бежичних комуникационих система. Развој софтверског алата који би на основу унетих параметара вршио предикцију остварености услова за појаву болести у многоме би смањио проблем одређивања правог времена примене хемијских третмана. Сами процес предикције заснива се на примени data mining техника, док се процес прикупљања и слања вредности метеоролошких параметара заснива на примени сензорних техника и бежичних комуникационих система. Data mining је област која се бави откривањем нових и потенцијално корисних информација из велике количине података [5]. Data mining помаже у откривању важних информација и знања утканих у податке, пословање и науци. Применом data mining техника над великим скупом података могу се уочити везе између података, логичности, као и шаблони у структурама података. Data mining се базира на два примарна задатка, а то су предикција и дескрипција. Задатак предиктивног data mining-а је да на основу међусобних зависности између познатих података предвиди догађај или вредност неке промењљиве која није позната. Дескриптивни data mining са друге стране се фокусира на проналажење шаблона који би описали међусобну везу између података на начин који је људима разумљив. Примена data mining техника показује да су ове технике веома често много снажније, флексибилније и ефикасније у домену истраживачке анализе података од статистичких техника[6]. Њихова примена у области пољопривредне производње је област која се веома брзо развија. Велики број data mining техника се може применити на проблеме са којима се савремена пољопривредна производња сусреће. Само неке од техника су неуронске мреже, стабла одлучивања, *support vector machine* (SVM), технике кластеризације и класификације, итд.

Павилним одабиром метеоролошких параметара и параметара који указују на пристуство спора у активном или пасивном стању може се креирати предикциони модел који ће на основу утврђене зависности између ових параметара вршити предикцију појаве конкретно одабраних болести.

Тачност предикционог модела зависи од величине тренинг скупа података као и од примењене data mining технике. Потврда тачности остварене предикције може се вршти помоћу података из тренинг скупа или помоћу новог скупа података који нису раније коришћени у процесу тренинга модела. У склопу овог истраживања коришћена су два скупа података. Ово практично значи да је један скуп података коришћен за креирање и тренирање предикционог модала, док је други скуп података коришћен у процесу евалуације креираног предикционог модела. Применом овако креираног предикционог модела у реалним условима број инстанци у тренинг скупу података се може повећати додавањем нових инстанци након сваке потврђене предикције. На овакав начин могу се пратити промене у понашању предикционог модела у зависности од уочених промена под којима може доћи до инфекције. Такође успешна предикција одговарајућег степена тачности отвара могућности примене креираног софтверског решења за предикцију других биљних болести. У већини случајева могао би се користити већи део једног те истог тренинг скупа података (метеоролошких података) док би се разликовале независне и зависне променљиве у предикционом моделу као и њихова међусобна зависност.

Како је процентуално мали број истраживања са тематиком примене data mining метода у воћарској производњи доступан, акценат овог истраживања је стављен управо на примену data mining техника и сензорних бежичних комуникационих система у воћарској производњи. Поменуте технике примењене су са циљем креирања софтверског решења које би након извшеног уноса или прикупљања података вршило предикцију остварености услова за појаву две болести које причињавају велике економске штете над воћним врстама. Патогени који изазивају ове две болести су *monilinia laxa* і *coccomyces hiemalis*. *Monilinia laxa* се јавља широм света у условима умерене и суптропске климе. Распросрањена је у свим воћарским рејонима Европе изузев крајњег севера. *Monilinia* се у Србији јавља сваке године, пре свега као паразит цветова, грана, гранчица, али и као проузроковач трулежи код коштичавог воћа. У Србији је *monilinia laxa* посебно значајна као паразит коштичавог воћа (кајсија, бресква, вишња, трешња, шљива и тд.), али се може јавити и код јабучастог воћа. Последица напада, посебно осетљивих сората наведених врста, је масовно

пропадање цветова и сушење гранчица, а након неколико година узастопног напада узрочника болести долази до пропадања грана, делова крошње и на крају сушења целих стабала. Осим тога, гљивица у дозревању узрокује и трулеж плодова. Уколико се посматра трулеж плодова највећи губици од ове болести настају у воћњаку и при транспорту. Губици у воћњаку могу бити 50% па чак и 75% а у току транспорта од 5 до 25%. Повећани губици настају у годинама обилних киша јер оне погодују овој гљиви и то може знатно умањити приносе. *Coccomyces hiemalis* проузрокује у народу познату болест под називом мрка пегавост листа. Најзаступљенија је код вишње и трешње и једна је од најопаснијих болести јер изазива превремено опадање листа током лета, које може довести до поновног цветања воћака у текућој сезони. Уколико до цветања и не дође, род је у наредној години је изузетно умањен јер биљка не може добро да се припреми за зимски период с обзиром да није имала лист којим би хранила себе и родне пупољке. Поред специфичних метеоролошких услова који морају бити задовољени да би дошло до појаве ове две болести, као и већине других, карактеристичан је и период године у коме се болести јављају. Период године везан је за фенофазу развоја у којој се биљка тренутно налази. Односно, под истим метеоролошким условима, уколико се биљка не налази у одговарајућој фенофази развоја, што одговара конкретном периоду године, неће доћи до појаве болести.

Реализација наведеног истраживања оријентисана је кроз две групе циљева. Прва група обухвата анализу и обраду података применом data mining техника кроз реализацију следећих циљева:

- Анализа метеоролошких и просторно-временских скупова података применом математичке регресије, класификације и визуализације података;
- Креирање предикционих модела на основу скупа података са познатим исходом применом клафикационих алгоритама и стабла одлучивања;
- Поређење тачности креираних предикционих модела применом већег броја евалуационих техника;
- Израда софтверског решења погодног за коришћење у реалним условима.

Анализа и mining великих просторно-временских података укључује припрему и прилагођавање података, обраду података и постпроцесирање. Просторно-временски подаци садрже стање неког објекта, догађаја или процеса у простору и времену. Овакве податке је могуће прикупити на различитим локацијама, у току различитих временских периода и у различитим форматима. Њихова сложеност се огледа у чињеници да садрже дискретне репрезентације континуалног простора и времена. Просторне податке је могуће категоризовати у три класе: објекти, поља и просторне мреже. Просторно-временски подаци се у зависности од начина репрезентације временске компоненте могу описати следећим моделима: модел стања у одређеном временском тренутку, модел промена стања у времену, модел догађаја и процеса [7]. Сваки ентитет у скупу просторно-временских података описује се торком атрибута. Могуће је издвојити три различита типа атрибута просторно-временских података: атрибути који не садрже нити просторну, нити темпоралну компоненту (ови атрибути се често зову тематски атрибути), просторни атрибути и темпорални атрибути. Тематски атрибути се користе за репрезентацију особина које су изван просторно-временског контекста. Ови атрибути имају исте особине као атрибути просторно и временски нереференцираних података. Просторни атрибути користе се за представљање геолокације (на пример, географске ширине и дужине), просторног опсега (површина, радијус), топологије и елевације у датом просторном референтном систему. Привремени атрибути приказују временске маркере који се односе на просторни објекат, растер или на стање просторне мреже, као и време трајања процеса. Особина просторно-временских података да поседују имплицитне везе између различитих типова атрибута (просторних и временских на пример) намеће потребу за развојем метода и техника за представљање тих релација. Један приступ у руковању имплицитним просторно-временским релацијама између атрибута јесте да се исте конвертују у колоне како би се омогућила даља обрада података, међутим овакав приступ може довести до губитка информација. Просторна и временска неодређеност, која је јако честа појава када се ради о просторно-временским подацима, може креирати различите проблеме везане за моделовање и обраду података. Преферирани приступ за руковање просторно-временским подацима јесте развој статистичких модела и техника за

инкорпорисање просторних и временских информација у сам процес анализе истих. Овај приступ уствари представља главну идеју која стоји иза области анализе и mining-а просторно-временских података [7]. Група метеоролошких података садржи информације о конкретним климатским параметрима. Ово су параметри до којих се долази свакодневним мерењем или мерењем више пута у току дана. Тачност ових параметара зависи од покривености конкретног подручја адекватним метеоролошким станицама. Уколико посматрано подручје није адекватно покривено или се налази у граничном појасу покривености једне станице, тачност метеоролошких података може бити умањена. Ови подаци су најчешће нумеричког типа и могу се процесирати различитим статистичким анализама и data mining методама. Како би се отклонили сви недостатци потребно је анализирати скуп података који садржи ове податке. Најчешћи недостатци метеоролошких података јесу невалидне вредности или недостатак вредности за неки од параметара. До ових недостатака долази када током процеса очитавања података неки од сензора престане радом, или пак дође до грешке приликом уноса података који датирају и неког прећашњег периода.

На основу метеоролошких, просторно-временских и података о појави поменутих двеју болести креиран је скуп података са познатим исходом. Овај скуп података након анализе и отклањања *outlier*-а користи се као тренинг скуп у процесу креирања предикционог модела. У процесу креирања предикционог модела циљ је користити методе надгледаног машинског учења. Технике надгледаног машинског учења налазе своју примену у многим областима. Надгледано машинско учење је процес учења скупа правила из примера који се налазе у тренинг скупу, стварајући као резултат процеса класификатор који може бити коришћен у процесу предикције на основу података из нових инстанци [8]. У надгледаном машинском учењу улаз у алгоритам је обично представљен у форми сета инстанци, где је свака инстанца представљена одређеним бројем атрибута за обучавање и ознаком класе (класним атрибутом). Инстанце података се у рачунару представљају у облику који је погодан за примену алгоритама учења. Код алгоритама машинског учења најпогоднији начин за представљање инстанци је помоћу неких њихових својстава, односно атрибута (*eng. feature, attribute*).

Та својства или атрибути представљају карактеристике инстанци, тако да сваки од изабраних атрибута може имати вредност која припада неком унапред задатом скупу. Вредности атрибута су често нумеричке, али могу бити и категоричке, односно могу представљати имена неких категорија којима се не могу једнозначно доделити смислене нумеричке вредности или значење. У машинском учењу, када су изабрани атрибути помоћу којих се инстанце описују, онда се свака инстанца може представити вектором вредности атрибута које јој одговарају [9]. Задатак алгоритма је да научи како да новој, необележеној инстанци података додели тачну ознаку класе. Уколико је вредност номинална ради се о класификацији, а уколико је нумеричка ради се о регресији. Сам квалитет добијеног класификатора се потврђује резултатима насталим тестирањем класификатора на неозначеним инстанцама (инстанце којима су познате вредности предикторских атрибута, али је непозната вредност ознаке класе). С обзиром да тестирање наученог знања над подацима на основу којих је учено обично доводи до значајно бољих резултата од оних који се могу касније добити у применама, потребно је пре употребе проценити квалитет наученог знања. Ово се обично постиже тако што се разматра колико је научено знање у складу са неким унапред датим подацима за тестирање. Тест скуп чине подаци за тестирање. Тест скуп треба да буде такав да је дисјунктан са тренинг скупом [9].

Поред различитих класификационих алгоритама, предикциони модели на основу тренинг скупа креирани су и помоћу стабала облучивања. Проблем креирања предикционог модела на основу стабла одлучивања решава се постављањем пажљиво састављених питања о атрибутима из тренинг скупа. Сваки пут се, по добијању одговора, прелази на следеће питање, док се не дође до закључка о вредности класног атрибута. Стабло одлучивања је хијерархијска структура која се састоји од чворова и директних грана [10]. Класни атрибути се налазе у листовима стабла. Код овако креираног предикционог модела процес одређивања вредности класног атрибута почиње од кореног чвора и поређењем вредноси атрибута у критеријуму поделе долази се до вредности класног атрубута који се налази у листу стабла. У пракси постоји много стабала одлучивања која се могу креирати на основу тренинг скупа. Нека су прецизнија од других, али се, с

обзиром на експоненцијалну величину простора, конструисање своди на креирање оптималног стабла.

Како би се пронашао предикциони модел са највећим степеном тачности, вршено је поређење тачности креираних предикционих модела применом већег броја евалуационих техника. Циљ евалуације креираних модела је утврдити колико класификација података креираним моделом одговара стварној класификацији. У зависности од начина посматрања перформанси креираних модела постоји више различитих мера за њихову евалуацију. У зависности од карактеристика посматраног проблема и начина његове примене, врши се избор најпогоднije мере. При евалуацији класификацијских модела основни појам је појам грешке. Уколико примена класификацијског модела на изабраном примеру доводи до резултата прогнозе класе која је различита од стварне класе примера, онда је настала грешка приликом класификације. Ако је свака грешка подједнако значајна, тада је укупан број грешака на посматраном скупу примера добар индикатор рада класификацијског модела. Једна од примењених техника у склопу истраживања је и унакрсна валидација (енг. *Cross Validation)* са одређеним бројем *fold-ova.* Применом ове технике тачност предикционих модела се пореди на основу низа грешака, као и процентуално коректно класификованих инстанци података. Предикциони модел са процентуално највећим степеном успешно класификованих инстанци и најмањим степеном грешке може се узети за коначни предикциони модел који ће се користити у реалним условима.

Обрада и анализа метеоролошких и просторно-временских података, као и креирање и евалуација предикционих модела, само су део функционалности имплементираних у оквиру софтверског решења чији је главни задатак коришћење у реалним условима. Софтверско решење представља крајњи циљ овог истраживања и треба да обувати поменуте функционалности, предикцију остварености услова за појаву болести и предикцију времена хемијских третмана. Предикција остварености услова за појаву конкретних болести, као и времена хемијских третмана, базира се на креираном предикционом моделу и новом скупу података који не садржи вредности класног атрибута. Скуп података за који је потребно извршити предикцију далеко је мањи од тренинг скупа и обично садржи

неколико инстанци. Саме инстанце података структурно су исте са инстанцама података у тренинг скупу података и садрже најновија мерења метеоролошких параметара, као и просторно-временске податке.

Покретањем предикције корисник софтверског решења добија могућности утврђивања остварености услова за појаву болести воћа узроковане развојем два претходно поменута патогена. Са сваком коректно класификованом инстанцом и са сваким тачно одређеним класним атрибутом инстанца новог скупа података може се преместити у тренинг скуп података, што води квалитетнијем предикционом моделу. Поред поменутих функционалности које су примарни циљ софтверског решења, софтверско решење треба да буде довољно интуитивно за коришћење како би га крајњи корисници (пољопривредни произвођачи и агрономи) могли примењивати у реалним условима.

Друга група циљева обухвата истраживања у пољу сензорне технологије и преноса сигнала кроз реализацију следећих циљева:

- Креирање модела метеоролошке станице са удаљеним приступом;
- Поређење карактеристика потребних сензора доступних на тржишту у циљу одабира најадекватнијих;
- Моделирање преноса очитаних вредности од метеоролошке станице до базног рачунара;
- Моделирање утицаја федингa на бежични пренос очитаних вредности;
- Моделирање утицаја федингa на GPS сигнал и слање података помоћу овог сигнала.

Прикупљање метеоролошких података од изузетног је значаја за креирање тренинг скупа података и обуку предикционог модела. Такође, аутоматизовано прикупљање ових података омогућава свакодневно вршење предикције остварености услова за појаву биљних болести. У зависности од локације на којој се метеоролошка станица налази, као и у зависности од периоде очитавања параметара постоји могућност предикције и више пута у току дана. Набавка и коришћење метеоролошких станица доступних на тржишту изискује велике економске издатке, како за пољопривредне произвођаче тако и за правна лица.

Из тог разлога, један од циљева истраживања у овом раду било је и креирање модела метеоролошке станице са удаљеним приступом. Модел метеоролошке станице заснован је на рачунарским компонентама и сензорима доступним на тржишту.

У циљу креирања што квалитетније метеоролошке станице вршено је поређење карактеристика потребних сензора. Карактеристике потребних сензора сагледане су са аспекта мерног опсега, напајања, утрошка енергије, као и отпорности на утицаје спољашње средине. Такође, код одабира сензора водило се рачуна о њиховој компатибилности са рачунарском компонентном на коју ће бити повезани. С обзиром да се ради о креирању метеоролошке станице чија је намена мерење метеоролошких параметара значајних за пољопривредну производњу, такве метеоролошке станице морају се поставити на производној површини. Постављање метеоролошких станица на производним површинама доводи до потребе имплементације удаљеног приступа и слања очитаних вредности метеоролошких параметара без одласка на конкретну локацију. У циљу реализације овог проблема вршено је поређење различитих могућности преноса сигнала које нуде бежични телекомуникациони системи. Поређење реализације удаљеног приступа помоћу радио канала и помоћу *GSM* мреже представља постављени циљ. У оба случаја слање података се обавља бежичним путем, те је утицај фединга на квалитет преноса и удаљеност метеоролошке станице од базног рачунара пресудан. Одабир адекватног бежичног телекомуникационог система вршен је моделирањем система у оквиру софтверских алата за симулацију бежичног преноса.

Последњи циљ у оквиру ове групе јесте моделирање преноса просторних података. За потребе прикупљања просторних података користи се *GPS* систем којим су одређени просторни параметри локације сваке од метеоролошких станица, њихова међусобна удаљеност, као и област покривености. *GPS* је скраћеница од *NAVSTAR GPS*. То је акроним од *NAVigation System withTime And Ranging Global Positioning System*. *GPS* је сателитски базиран систем који користи констелацију од 24 сателита да кориснику пружи тачну позицију.

GPS је оригинално пројектован за употребу од стране војске, како би радио било где на површини Земље. Убрзо након остваривања оригиналних намена, постало је јасно да и цивили могу користити овај систем, и то не само за лично позиционирање. Прве две велике цивилне примене које су извршене биле су маринска навигација и премер. Данашње примене се крећу од навигације унутар аутомобила, преко управљања конвоја камиона, до аутоматизације грађевинске машинерије.

GPS конфигурација се састоји од три различита сегмента: свемирски сегмент, кога чине сателити који круже око Земље, контролни сегмент, кога чине базне станице позиционоране на Земљином екватору за контролу сателита и кориснички сегмент, кога чини било који пријемник који прима и користи GPS сигнал. Постоји више различитих метода за добијање позиције користећи GPS. Употребљени метод зависи од захтеване тачности и типа раположивог GPS пријемника. Моделирање утицаја фединга на пренос GPS параметра вршиће се применом симулације у неком од алата за симулацију, у зависности од одабраног GPS пријемника. Одабир самог пријемника условљен је функционалностима истог и начином повезивања са одабраном рачунарском компонентом на којој је базирана метеоролошка станица.

Наведени циљеви, груписани у две групе различите по примењеним технологијама и очекиваним резултатима, заједно креирају оквире који воде ка реализицији предмета спроведеног истраживања. Сами предмет истраживања, као и захтеви реализације, врше обједињавање информационо комуникационих технологија и бежичних комуникационих система са циљем осавремењавања пољопривредне производње. У складу са тим, спроведено истраживање може се сматрати мултидисциплинарним. Може се очекивати интезивна примена финалног продукта, у виду софтверске апликације, у процесу одређивања времена хемијских третмана.

3. Преглед досадашњих истраживања

У једном од радова аутори су развијали систем за даљинско очитавање метеоролошких података [11]. Главни циљ система је удаљени приступ на неприступачним теренима, као и приступачна цена реализације. Помоћу *data logger-a* подаци су очитавани сваког пуног сата. Аутоматски процес успостављања интернет конекције путем *USB* модема се покреће након очитавања. Очитани подаци се шаљу на удаљени сервер и конекција се прекида. Даља обрада података врши се на серверској страни, као и чување у бази података. Преглед вредности очитаних података омогућен је помоћу *web* интерфејса. За конкретно мерење метеоролошких података искоришћена је *Davis Vantage Pro2* бежична метеоролошка станица, опремљена сензорима за количину падавина, температуру, влажност ваздуха, брзину ветра и сензором за мерење *UV* зрачења. Напајање је обезбеђено путем соларног панела и једне батерије од 3 V. Овакво напајање омогућава континуалан рад до две године, док је животни век батерије без соларног панела процењен на 8 месеци. Комуникација и слање података обезбеђено је помоћу *USB HIT 39* модема, чији се рад базира на употреби *SIM* картице.

Систем за мерење тренутних метеоролошких вредности заснован на употреби *Adruino* физичко-рачунарске платформе отвореног кода фирме *Freaklabs* и сета метеоролошких сензора описан је у [12]. Сет одабраних сензора врши мерење температуре, влажности ваздуха, интезитета светлости, тачке росе и индекса топлоте. За мерење температуре искоришћена су два сензора *DHT11* и *LM35*. Сензор *DHT11*, поред мерења температуре, врши и мерење влажности ваздуха. Тачка росе, као и топлотни индекс прерачунавани су на основу очитаних вредности температуре и влажности ваздуха. Други температурни сензор *LM35* врши мерење само температуре. Мерни температурни опсег овог сензора износио је од -55^{o}C до 150^{o}C. Светлосни сензор искоришћен је за мерење интензитета светлости. Предложени систем, поред мерења тренутних вредности метеоролошких параметара, користи се и у циљу предикције будућих временских услова. На основу графичког представљања прикупљених података креиран је метеоролошки шаблон за дати период времена.

Овакви шаблони могу указати на специфичности метеоролошких услова над неким конкретним подручјем, и то графичким поређењем метеоролошких вредности, тренутно очитаних са оним прикупљеним у прошлости.

Праћење тренутних метеорлошких вредности, базирано на бежичној мрежи сензора и *Adruino* платформи, искоришћено је у циљу креирања повољних метеоролошких услова у стаклeнику [13]. Рад система базиран је на употреби две *Adruino* платформе и сета сензора. На једној од платформи повезани су сензори за мерење температуре, влажности ваздуха, влажности земљишта, количине светлости, као и количине угљен-диоксида у простору. За мерење температуре и влажности ваздуха искоришћен је *DHT11* сензор. Мерење влажности земљишта обезбеђено је помоћу *SEN92355P* сензора фирме *Grove technology*. Проценат угљен-диоксида у ваздуху мерен је помоћу *FC-22-1* сензора, док је количина светлости мерена помоћу фото сензора. Са ове платформе слање измерених вредности података вршено је помоћу *Chibi Wi-Fi-a* и *GSM* модула до друге платформе за обраду података, као и до мобилног телефона корисника. *Wifi* технологија је искоришћена за растојање до 100 метара и претежно служи за комуникацију између већ поменуте две *Adruino* платформе. Слање пакета података путем *GSM* модула обезбеђено је помоћу *SIM900 Quad-band GSM/GPRS* хардвера који омогућава слање података на широком географском подручју. Слање очитаних података и издавање наредби за регулацију метеоролошких услова унутар стаклeника организовано је путем *SMS*-а. Подаци примљени на другој *Adruino* платформи у виду *SMS*-а, обрађују се и приказују кориснику у оквиру *GUI* (енг. *Graphical User Interface*) панела одакле се врши издавање наредби.

Систем за доношење одлука, базиран на употреби метеоролошке станице интегрисане у оквиру система бежичне мреже сензора, развијен је у циљу унапређења прецизне пољопривредне производње и приказан је у [14]. Развијена метеоролошка станица базира се на раду шест сензора и соларном напајању. Интегрисани сензори могу мерити брзину ветра, смер ветра, температуру ваздуха, соларно зрачење и количину падавина. Брзина ветра и смер ветра мерени су са *ADS* анометром и показивачем правца. Сензор за мерење брзине ветра ради у

опсегу од 4 km/h – 201 km/h. Сензор за мерење количине падавина је креиран као посуда која се сама празни и креира прекид на микроконтролеру са сваких 0,2794 mm кише. За мерење температуре ваздуха и релативне влажности ваздуха коришћен је *SHT15* сензор. За потребе мерења соларног зрачења коришћен је соларни сензор, који помоћу фотодиоде (*BPW21*) конвертује соларно зрачење у електричну струју, а касније помоћу *AD* конвертора на микроконтролеру сигнал се конвертује у дигитални. Непрекидно напајање овако креиране метеоролошке станице засновано је на два соларна панела који врше пуњење батерије. Слање очитаних података до сервера вршено је помоћу *GPRS* модема. Развијена метеоролошка станица постављена је на пољу маслина у оквиру Пољопривредног института Tunisie (Северна Африка). Предност овако креиране метеоролошке станице, поред аутоматског очитавања и слања вредности конкретно потребних параметара, је и нижа цена реализације у односу на комерцијалне аутоматске метеоролошке станице.

Метеоролошка станица развијена и постављена на *Edamame* фарми поређена је са комерцијалном метеоролошком станицом *Davis Vantage Pro2* [15]. Систем сензора за мерење метеоролошких параметара обухватио је мерење температуре и влажности ваздуха, светлости, брзине и правца ветра, количине падавина, температуре и влажности земљишта. Сви сензори су добављени од стране *Warf Corporation Co., Ltd*. компаније. Сензори за мониторинг метеоролошких параметра су постављени на 3 m изнад земље, док су сензори за праћење температуре земљишта и влажности земљишта постављени у земљиште на 10 cm и 25 cm дубине, респективно. Модул за прикупљање и слање података заснован је на *PIC24FJ64* микроконтролеру који представља централну процесорску јединицу. За слање података од микроконтролера до базне станице коришћен је *ZigBee* модул *Xbee Pro S2* серије *MaxStream* компаније. Карактеристично за овај модул је то да се базира на IEEE802.15.4 стандарду, као и да му је домет у линији прегледности до 1,6 km. За централну јединицу базне станице одабран је *Raspberry Pi 2* модел *B*. Са овог рачунара путем модема аутори су креирали интернет конекцију која служи за складиштење очитаних вредности параметра на *cloud*. Статистичким поређењем вредности очитаних параметара са креиране метеоролошке станице и *Davis Vantage Pro2* метеоролошке станице, аутори су

утврдили да није било великих одступања, што доводи до закључка да овако креирана сопствена метеоролошка станица намењена употреби у пољопривреди обезбеђује скоро истоветни степен тачности мерења уз мање економске издатке.

Систем за праћење метеоролошких параметра у реалном времену, заснован на *ZegBee* бежичној мрежи сензора, постављен је и коришћен у склопу истраживања током годину дана на 40 хектара *GranMonte* винограда [16]. Систем се састојао од пет мобилних метеоролошких станица, посебно направљених за ову намену, и повезаних са централним рачунаром за складиштење података. Рад сваке од станица заснован је на *PIC18F45J10* микроконтролеру фирме *Microchip*, сензорима за мерење температуре и влажности ваздуха, *ZigBee* модулу за *wifi* слање података, соларном панелу и јединици за напајање. *SHT15* сензор је искоришћен за мерење температуре и влажности ваздуха. Овај сензор заштићен је посебним цилиндричним омотом од спољашњег утицаја падавина и директне сунчеве светлости. Након сваког мерења микроконтролер је вршио слање података путем предајника. Након очитавања и слања у циљу уштеде енергије мерна станица прелази у мод мировања. Рад базне станице био је заснован на употреби PIC18F87J10 микроконтролера који располаже са 1Mb флеш меморије и 2 *UART* (енг. *Universal Asynchronous Receiver/Transmitter*) модула. Микроконтролер је примао податке са терена путем *ZigBee* мреже. Након пријема, подаци са сваког чвора су памћени у меморији микроконтролера и слати до рачунара на свака два минута. Рачунар је у овом систему радио као *data logger* како би се омогућило чување велике количине података у циљу анализе истих, као и анализе понашања винове лозе у конкретним временским условима.

Систем за мерење вредности тренутних метеоролошких параметара заснован на уграђеним системима, *crowdsourcing*-u и телеком инфраструктури приказан је у [17]. Рад мерне станице задужене за мерење тренутних вредности базиран је на *Arduino Uno* и *LP1768* микроконтролерима. Сензори за мерење температуре и влажности ваздуха (*DHT22*), амбијенталног осветљења (*BH1750FVI*), количине падавина (*RM0998*), количине гасова у атмосфери (*RM1108*), смера и брзине ветра (*MD0550*) повезани су на поменуте микроконтролере. Овако креиране мерне станице постављене су на врх телекомових торњева за пренос сигнала мобилне

телефоније. На овај начин су мерења метеоролошких параметара ваздуха вршена у простору изнад зграда. Постављањем мерне станице на торањ мобилне мреже решен је проблем напајања и заштите од људског утицаја. Слање података до базне станице, након сваког мерења, вршено је уз помоћ *ZigBee Tx-Rx* трансмитера који добија податке од *Arduino* микроконтролера. Комуникација између микроконтролера и *ZigBee* трансмитера обезбеђена је помоћу *RS232* протокола. Базна станица била је опремљена ZigBee пријемником за пријем података и *GSM* модулом за њихово даље слање до корисника. Слање података на захтев корисника омогућено је путем *SMS*-а. На захтев корисника, који је пристигао путем *AT* команде, *GSM* модул је коришћењем *SIM* картице вршио слање података смештених у *EEPROM* меморији, повезаном са микроконтролером. Предности оваквог система наведене од стране аутора су лака проширљивост у домену пољопривреде (сензори за влажност земљишта и количину воде у земљишту), као и ниска цена реализације која износи око 100 долара.

Аутоматско прикупљање и пренос тренутних вредности метеоролошких параметара са већег броја удаљених локација било је предмет истраживања групе аутора [18]. Систем је заснован на употреби *TINI* микроконтролера, као и сензора за мерење тренутних вредности температуре ваздуха, влажности ваздуха и правца ветра. *TINI* микроконтролер има подршку за *TCP/IP* протокол који је искоришћен за потребе удаљеног приступа. Опсег мерења температурног сензора повезаног са микроконтролером помоћу једног проводника био је између -55 ^{o}C до +125 ^{o}C, док је тачност овог сензора у границама ±0.5 ^{o}C. Сензор за мерење влажности ваздуха *HC3223* био је отпоран на утицај хемикалија и воде из спољашње средине, а опсег мерења релативне влажности ваздуха био је од 0 до 99%. Сензор за одређивање смера ветра састојао се из осам преклопника, од којих је сваки положај одређивао по једну од могућих страна света. Путем *GUI*-а креираног у *HTML*-у вршен је приказ тренутно очитаних вредности мерења кориснику. У исто време, вршено је и слање очитаних вредности у централну базу података ради даљег чувања.

Студија изводљивости спроведена над системом за прикупљање метеоролошких података и података о пристуству биљних штеточина заснованом на употреби *GSM-SMS* комуникационе архитектуре описана је у [19]. Аутори су креирали формат пакета за пренос податка у виду кратких порука које су погодне за мониторинг обрадиве површине и прикупљања података, као што су температура, влажност ваздуха, брзина ветра и број уловљених штечина/инсеката. *GSM-SMS* комуникациона архитектура за пренос текстуалних података одабрана је за потребе бежичне везе унутар системске комуникације из више разлога. Један од разлога је и могућност репрезентације очитаних нумеричких вредности у текстуалном формату. Још један од разлога је и значајна улога *SMS*-а у преносу текстуалних порука у оквиру 3G ере. Издвојене су три предности *SMS*-а над осталим технологијама приликом употребе у оваквим системима. Као прва од предности истиче се употреба бежичне везе над комуникацијом оствареном жичаним повезивањем. Заштита *SMS* података остварена енкрипцијом коју нуди *GSM* мрежа друга је од предности. Као трећа предност истиче се ретрансмисија у случају отказа прликом преноса података, која се остварује подешавањем ретрансмисионих тајмера. Имплементација протипа овог система била је заснована на два подсистема. Главна карактеристика оба подсистема била је употреба *GSM* бежичне мреже за слање података. Први подсистем састојао се од *Texas Instrument (TI) 16-bit RISC* микропроцесора *MSP 430-F449* који је представљао кернел и на који су повезани GSM модул, GPS модул, модул за праћење временских параметра и модул за праћење пристуства инсеката. Овај подсистем назван је платформом за мониторинг (енг. *FMP-Field Monitoring Platform*). Кернел модул користио је два универзална синхроно/асинхрона трансмитера (енг. *USART – Universal Synchronous and Asynchronous Receiver and Transmitter*) за *RS232* комуникацију са свим сензорима за мерење вредности спољашњих параметра и прикупљање података. Дигитални сензор *AM-4205*, компаније *Lutron Electronics* коришћен је за мерење брзине ветра, температуре и влажности ваздуха. Опсег мерења брзине ветра био је од 0,4 m/s до 25,0 m/s са тачношћу од ±2%. Опсег мерења влажности ваздуха био је од 10% до 95%, док је тачност била ±1% за влажност ваздуха ≥70%, а ±3% за влажност ваздуха < 70%. Температура ваздуха мерена је у опсегу од 0^{o}C do 50^{o}C, са тачношћу од ±0,8^{o}C.

Поред поменутих сензора микропроцесор је повезан и са електронским хватачем инсеката. Електронски хватач инсеката био је обложен феромоном који је привлачио мужјаке мољаца. Приликом сваког контакта инсекта са хватачем, бројач уловљених инсеката се инкрементирао. Други подсистем (енг. *HCP – Host Control Platform*) путем GSM модула примао је прослеђене податке од стране *FMP-e*. Након пријема података вршено је њихово декодирање и смештање у базу података у циљу њихове будуће обраде. Овај подсистем био је замишљен тако да у случају када се на основу прикупљених података детектује аномалија (повећано присуство инсеката) систем активира упозорење које се саље администратору система, фармеру или некој другој организацији. С обзиром да је садржај *SMS* поруке текстуални, аутори су унапред дефинисали формат поруке која је служила за пренос информација од *FMP-a* до *HCP-a*. Максимална могућа величина сваке прослеђене поруке била је 160 бајтова. Првих 5 бајтова чинили су заглавље поруке, док су преостала 155 бајта коришћена за пренос података. Перформансе *GSM-SMS* архитектуре имплементиране у оваквом систему тестиране су у склопу истраживања. Мобилни уређај *WAVECOM WMOD2B M1203A* са *SIM* картицом инсталиран је и коришћен на оба подсистема (*FMP i HCP*). Тест је спроведен коришћењем три различита мобилна оператера присутна на територији Тајвана. Аутентификациони тест је показао да су прикупљени подаци пренесени коректно. Тест перформанси система спроведен над 915 преноса података показао је да је за једнократно слање података од *FMP* до *HCP* било потребно око 10-15 s, док је просечно време одговора *FMP* на захтев *HCP* било око 30,5 s. Остварена тачност слања података путем *SMS*-а била је 100%. Стопа ретрансмисије износила је 2,73%, док је укупна стопа изгубљених података услед неуспеле конекције или ретрансмисије износила 0,66%.

Систем за праћење метеоролошких параметара и утицаја временских услова на бројност *Bactrocera dorsalis* (*Hendel*) инсекта описан је у раду [20]. Принцип рада система заснован је на два одвојена подсистема слично као у раду [19]. Први део представља платформу за прикупљање података *(*енг. *RMP-Remote Monitoring Platform*), док други део (енг. *HCP-Host Control Platform*) представља платформу за складиштење и статистичку обраду прикупљених података. Рад *RMP*-а заснован је на микроконтролеру *MSP430F449* развијеном од стране *Texas*

Instruments, Inc. 2006. године и представља главни процесорски чип. Овај чип служи за креирање и слање пакета података, као и за пренос контролних команди између модула који се користе у *RMP*. Поред микропроцесора сваки од *RMP*-а био је опремљен и анемографом *AM-4203* компаније *Lutron*, чија је тачност износила ±2,0%. Овај анемограф поседује *RS232* серијски порт и пружа 16-битни излазни сигнал помоћу кога комуницира са микропроцесорским чипом. Сензор одабран за симултано мерење температуре и влажности ваздуха био је *SHT75* компаније *Sensirion*. Овај сензор обезбеђује дугорочну стабилност и степен тачности од ±2,0% за релативну влажност ваздуха и ±0,4^{o}C за температуру. *GSM* модул *FASTRACK M1203A* компаније *WAVECOM Corporation* коришћен је, како у оквиру *RMP* тако и у оквиру *HCP*. Изабрани GSM модул испуњава *GSM900* и *GSM1800* спецификације. GPS ресивер *GM44* компаније *San Jose Navigation Corporation* са RS232 интерфејсом коришћен је у овом систему. Његова улога била је одређивање географске локације *RMP*-а, са тачношћу до 15 метара. Прикупљени метеоролошки подаци, подаци о броју уловљених инсеката, као и GPS локација *RMP*-а, преношени су у виду *SMS* поруке помоћу GSM модула од *RMP* до *HCP*. Овако креирани систем вршио је прикупљање и слање података аутоматски. *HCP* подсистем садржао је *GUI* који је коришћен за контролу преноса података од GSM модула. *GUI* је линкован са *MySQL* базом података у којој се чувају прикупљени теренски подаци који долазе са *RMP*-а. На *HCP* успостављен је *Apache* сервер на коме су кретиране web стране за преглед података, њихову статистичку анализу и репрезентацију у облику дијаграма.. Процес комуникације између *RMP* и *HCP* је био заснован на провери сваког пакета података на пријемној страни. У случају грешке, *HCP* је дизајниран тако да шаље захтев за ретрансмисијом, на основу кога ће *RCP* поновити слање података. Даља обрада података и алармирање корисника уколико је број инсеката изнад предефинисане вредности обавља се од стране *HCP*. Пре пуштања система у рад аутори су вршили тестирање исправности рада сваке од појединачних компоненти. Тестирање рада система трајало је 304 дана, у периоду од 1.7.2006. године до 30.4.2007 године. У процес тестирања биле су укључене две *RMP* платформе које су вршиле мониторинг и слање података до заједничког *HCP*. Тестови су показали да је просечна грешка *GPS* система износила 3,01 m, док је вредност

максималне и минималне грешке износила 22,22 m и 0,13 m респективно. Током овог периода процес слања података помоћу *GSM* модула протекао је без евидентираних губитака пакета података.

Систем за праћење агроеколошких услова на производним воћарским површинама Тајвана заснован на платформама са удаљеним приступом креиран је у склопу једног од истраживања [21]. Архитектурно целокупан систем био је подељен у три слоја. Први слој система је био задужен за мониторинг метеоролошких услова и евиденцију бројности инсеката. Други слој је био телекомуникациони слој који је коришћен за пренос података од платформе за мониторинг до модула за обраду података. Трећи и последњи слој представља слој на коме су подаци складиштени и анализирани. Систем за мониторинг се састојао од 12 бежичних сензорних мрежа (укупно опремљених са 163 сензорних чворова) и три самосталне мониторинг станице. Овако креиран систем покривао је 20 пољопривредних локација. Главна компонента самосталних мониторинг станица био је *MSP430* микроконтролер (*MSP430FG4619*) компаније *Texas Instruments*. Овај микроконтролер за потребе мерења температуре и влажности ваздуха био је опремљен метеоролошким сензором *SHT71* компаније *Sensirion, Inc.* Тачност овог сензора према спецификацији износила је ±0,4 ^{o}C код мерења температуре и ±2% код мерења релативне влажности ваздуха. Поред овог сензора микроконтролер је био опремљен уређајем за аутоматско бројање ухваћених инсеката, *GSM* модулом *GM44* компаније *San Jose Navigation, Inc.* за слање података, *GPS* ресивером за одређивање локације, као и самосталним напајањем. Самостално напајање било је базирано на соларном панелу од 20 W и батерији капацитета 100 Ah. Код мониторинг станице засноване на бежичним сензорним мрежама рад сваког од бежичних сензорних чворова заснован је на *ZigBee* преносном модулу. *ZigBee* модул био је повезан са аутоматским хватачем инсеката, 8051 контролером за бројање инсеката, инфраред контролером за генерисање прекида, светлосним сензором, сензорима за мерење тренутних вредности температуре и влажности ваздуха, соларним панелом од 20 W, као и батеријом капацитета 36 Ah. Два типа подсистема, креирани као својеврстан *gateway*, прикупљали су очитане вредности са сензорних чворова и даље их прослеђивали до сервера. Gateway је такође вршио мерење метеоролошких

параметара коришћењем сопствених метеоролошких сензора. Подаци о очитаним тренутним метеоролошким вредностима, броју уловљених инсеката и локацији станице, организовани су у *SMS* и помоћу *GSM* модула прослеђивани до сервера сваких 30 минута. Сви подаци су чувани у *MySql* бази података ради даље обраде. Стручњацима који су вршили анализу података обезбеђен је приступ подацима путем web сервиса креираних у *PHP*-у.

Метеоролошка станица, посебно дизајнирана за праћење одређених метеоролошких параметара, искоришћена је за прикупљање података потребних за рад система за доношење одлука примењеног у прецизној пољопривреди [22]. Архитектура овако креираног система била је подељена у два модула. Први модул заснован је на употреби *PIC18F2620* микроконтролера који комуницира са рачунаром и *Xbee* модулом за слање података путем серијског интерфејса. Слање очитаних података вршено је помоћу радио сигнала. Други модул састојао се од *RJ11* проширења на које су били повезани метеоролошки сензори и додатни уређаји за слање података до микропроцесора. Овај модул био је задужен за праћење температуре и влажности ваздуха помоћу *SHT75* сезора, сунчевог зрачења помоћу *Davis* сензора, количине падавина помоћу посуде са бројачем, као и брзине и смера ветра помоћу *ADS* аноментра и *ADS* ване уређаја респективно. Микропроцесор је био подешен тако да врши очитавање вредности на сваких 5 минута. Очитане вредности чуване су у *EEPROM* меморији. Након очитавања вршено је активирање *Xbee* модула који је обављао слање података, и потом се враћао у неактивни мод. Оваквом организацијом вршена је уштеда енергије с обзиром да је напајање система било обезбеђено помоћу батерија и соларног панела. Креирана метеоролошка станица тестирана је поређењем очитаних вредности са вредностима добијеним очитавањем са професионалне метеоролошке станице. Тестирање је показало да није било разлике у измереним вредностима. Овако креирани систем био је монтиран на производним површинама јабуке одакле је вршено тестрање процеса слања очитаних вредности радио каналом и њихова даља обрада у систему за доношење одлука.

Систем за аутоматско праћење метеоролошких услова са циљем прикупљања података ради даље обраде и доношења одлука описан је у раду [23]. Рад система

базиран је на *LPC1768 Cortex-M3* микроконтролеру. Микроконтролер је помоћу контролера и *ADC* пинова, који пружају могућност конверзије аналогног у дигитални сигнал, био повезан са сензорима за праћење тренутних вредности метеоролошких услова. Праћење метеоролошких услова је било базирано на мерењу тренутних вредности влажности ваздуха, температуре ваздуха, присуства гасова у ваздуху, као и брзине и смера ветра. Сензор за праћење влажности ваздуха радио је на принципу релативне влажности и генерисао је излаз у облику електричног напона. Са повећањем релативне влажности ваздуха импеданса се смањивала. *LM35* температурни сензор искоришћен је за мерење температуре. Излаз у облику аналогног електричног напона је био пропорционалан температури. Овај сензор није захтевао никакву спољашњу калибрацију, док је његова тачности износила ±0,4^{o}C на собној температури и ±0,8^{o}C уколико је температура у опсегу од 0^{o}C до +100^{o}C. *MQ-6* гасни сензор, креиран на бази SnO, био је јако осетљив на присуство пропана, бутана, *LPG*-а и природног гаса у ваздуху. Брзина и смер ветра праћени су помоћу класичног анометра. Као извор напајања искоришћен је соларни панел и батерија од 12 v. Вредности измерене помоћу поменутих сензора су након конвертовања у дигитални облик прослеђиване помоћу *RS232* серијске комуникације до *excel* табеле креиране у оквиру *LABVIEW* окружења. Такође, измерене вредности су у виду *SMS*-а помоћу *GSM* модула прослеђиване на мобилни телефон корисника. Овако креиран систем погодан је за примену у пољопривреди, како за праћење метеорлошких параметара тако и за контролу различитих система чије покретање зависи од тренутних метеорлошких вредности (фертигациони системи, сисеми за заливање и орошавање, као и системи регулације температуре ваздуха и нивоа штетних гасова у затвореним производним површинама).

Систем за даљинско праћење метеоролошких параметара, базиран на раду два одвојена модула, креиран је за потребе мерења температуре ваздуха, ваздушног притиска, влажности ваздуха количине падавина, брзине и смера ветра [24]. Опсег мерења одабраног температурног сензора био је од 0^{o}C до +50^{o}C, док је мерни опсег сензора за мерење влажности ваздуха износио од 30% до 90%. Максимална вредност атмосферског притиска коју је сензор за мерење притиска могао да измери износила је 110.800 паскала. Сензор за мерење количине падавина креиран

је на принципу посуде која се сама празни са сваких 0,02 mm падавина, креирајући импулс који је увећавао вредност бројача. Укупна количина падавина израчунавана је множењем вредности бројача са 0,02 mm падавина. Овакав принцип захтевао је ресетовање вредности бројача са сваким завршетком дана. За брзину и смер ветра као и у претходним истраживањима коришћен је анометар. Све вредности добијене са сензора су помоћу D1 мини контролера конвертоване у одговарајући формат потребан за слање. Након конвертовања помоћу бежичне мреже подаци су прослеђивани до рутера на сервер страни. Фреквенција *wifi* модула износила је 2,4 GHz. База података креирана за потребе чувања измерених вредности параметара креирана је у *MySql*-u. Подаци су са сервера помоћу PHP скрипте приказивани у оквиру web стране чији је садржај освежаван сваких 10 s.

Група аутора конструисала је метеоролошку станицу током чије реализације је један од главних циљева била ниска цена компоненти и очитавање података путем удаљеног приступа. Дизајн је заснован на три главна дела. Први део представља спољашњи модул који врши мерења, други део је унутрашњи модул који врши повезивање спољашњег модула са персоналним рачунаром, док је трећи део већ поменути персонални рачунар са софтвером за визуализацију и чување података [25]. Рад спољашњег модула је базиран на четири подсистема: јединица напајања, јединица са сензорима, микроконтролер и трансмитер радио фреквенције. Напајање је регулисано путем константног DC напајања и путем батерије која се користи у случају нестанка константног напајања. За мерење метеоролошких параметара (релативна влажност ваздуха, температура, атмосферски притисак и брзина ветра) аутори су се одлучили за три сензора у оквиру јединице са сензорима. *DHT11* сензор је искоришћен за мерење температуре и влажности ваздуха. Моторола *MPX4250A* сензор је коришћен за потребе мерења атмосферског притиска. Измерена вредност атмосферског притиска генерише се у форми аналогног сигнала (напона) који је пропорционалан измереном атмосферском притиску. Ефективни мерни опсег овог уређаја је између 0 и 250 kPa. Ова вредност одговара излазном напону у опсегу од 0,2 V до 4,9 V. Уколико је температура на којој ради овај сензор у опсегу од 0 ^{o}C до 85^{o}C, грешка приликом мерења је занемарљива. Због недоступности анометра електромотор без четкица је искоришћен као сензор за брзину ветра. Механички

механизам од три полулопте је искоришћен за потребе окретања мотора. Полулопте су постављене тако да окретање једне утиче на окретање друге па се на такав начин генератор окреће само у једном смеру и на тај начин генерише излазни напон константног поларитета. Излазни напон генератора мења се у складу са брзином ветра и доводи на улаз *AD* конвертора на микроконтролеру. За потребе преноса очитаних вредности искоришћен је *TLP* 315 радио фреквентни трансмитер који ради на фреквенцији од 315 MHz. Овај трансмитер користи *ASK* (енг. *Amplitude Shift Keying*) модулацију за пренос података. Трансмитер генерише сигнал од 315 MHz приликом слања бинарне 1, а не генериши никакав излаз када је на линији са подацима присутна бинарна нула. Микроконтролер који је коришћен за потребе рада метеоролошке станице је 8-битни *ATtiny48/88* контролер фирме *Atmel Corporation*. Ово је контролер са многим уграђеним периферијама и побољшањима у односу на традиционални 8051 контролер исте фирме. Уграђени серијски периферни примопредајник на овом контролеру коришћен је у комбинацији са радио фрекветним трансмитером приликом преноса података од спољашњег до унутрашњем модула. Такође *AD* конвертор микропроцесора је коришћен за конверзију аналогног сигнала добијеног са *MPX4250* сензора за мерење атмосферског притиска. Очитавање вредности са сензора вршено је сваког минута слањем *read* наредбе са микроконтролера. Очитане вредности су бежичним путем на већ поменути начин прослеђиване до унутрашњег модула. Унутрашни модул састојао се од два *ATtiny48/88* микроконтролера уклопљена у једну јединицу. Први микроконтролер био је идентификован као мастер и подешен тако да користи *USART* и серијски периферни интерфејс *SPI* (енг. *Serial Peripheral Interface*). *USART* је био задужен за пријем и проверу исправности података добијених из спољашњег модула. Провера је вршена због шума из атмосфере под чијим је константним утицајем радио фреквентни пријемник. Оваквом провером невалидни битови података су одбацивани, док су валидни подаци прослеђивани другом (*slave*) микроконтролеру путем *SPI*. Други микроконтролер примљење податке прослеђује путем *SPI* до серијског порта персоналног рачунара. С обзиром да персонални рачунар користи *RS232* за потребе серијске комуникације, док је комуникација на микроконтролеру базирана на *TTL*, разлика у њиховим радним

напонима превазиђена је употребом *MAX* 232 конвертора напона фирме *Maxim Integrated*. При свакој комуникацији овај конвертор је вршио конверзију напона са RS232 у *TTL* и обрнуто. База података била је реализована у *Microsoft Excel*-у док је *GUI* за потребе интеракције са корисником креиран коришћењем *MatLab* платформе. Овако креирана метеоролошка станица постављена је на две локације (*Mini Campus* и *Main Campus* Федералног Технолошког института, Minna, Nigerije) 15.9.2012. године и 9.12.2012. године, респективно. Аутори су добијене вредности мерења упоредили са вредностима добијеним са других инструмената. Поређење је показало да су вредности, добијене са креиране метеоролошке станице, у корелацији са вредностима добијеним са других мерних уређаја.

Аутоматска метеоролошка станица креирана за прикупљање метеоролошких параметра са специфичне локације, на бази удаљеног приступа, описана је у раду [26]. Ова метеоролошка станица састојала се од два модула: комуникационог и модула за обраду података. Задатак овако дизајниране странице је био мерење метеоролошких параметара попут температуре, влажности ваздуха, ваздушног притиска, брзине и смера ветра, количине падавина, сунчеве радиације, дужнине трајања дана као и мерење времена од изласка до заласка сунца. Аутори су одабрали *ATmega64* 8-битни *CMOS* микроконтролер за потребе централне процесорске јединице. Овај микроконтролер се карактерише малим утрошком енеригије, док у исто време може да изршава 16 милиона инструкција у секунди, што је еквивалентно брзини извршења од 192 MHz нормалног микроконтролера. У циљу успостављања комуникације овог микроконтролера и сензора искоришћен је 16-битни *AD* конвертор, како би се аналогни сигнал са сензора превео у дигитални. За потребе преноса података од микроконтролера до *host* рачунара искоришћена су два Siemens *TC35i* комуникациона модула, један на предајној и један на пријемној страни респективно. TC35i модул се може веома лако надоградити до *GPRS* модула који је компатибилан са *GSM* 2/2 +, као и *GSM900/GSM18000* и *RS232* портом за податке. Комуникација између микроконтролера и *TC35i* комуникационог модула остварена је преко серијског порта пропусног опсега 9600 Kbps. Комуникација је 8-битна, што практично значи да се увек преноси 8 битова података и 1 стоп бит. Комуникација између аутоматске метеоролошке станице и host рачунара била је креирана тако да се

може поделити у две категорије: периодични подаци и подаци по захтеву. Периодични подаци одговарају подацима који се у унапред дефинисаним временским интервалима прослеђују од аутоматске метеоролошке станице до *host* рачунара, док подаци по захтеву одговарају *host* рачунару и представљају упит послат метеоролошкој станици у циљу достављања података пре истека предвиђеног интервала. Пакет педиодичних података поред самих вредности метеоролошких параметара садржао је и временску компоненту у виду године, месеца, дана, сата, минута и секунде у којој је пакет послат. Сваки пакет података садржао је и јединствени код метеоролошке станице, на основу кога је на *host* рачунару вршено архивирање података за сваку станицу посебно како би се евидентирала разлика између метеоролошких параметара на различитим локалитетима.

За потребе једног од истраживања група аутора је креирала аутоматску метеоролошку станицу која је била постављена у граду Agrinio у западној Грчкој. Циљ овако креиране метеоролошке станице било је поређење очитаних вредности са вредностима добијеним са комерцијалних метеоролошких станица, постављених на аеродрому и ужем градском језгру. Метеоролошка станица била је опремљена сензорима за мерење температуре и влажности ваздуха, брзине и смера ветра, количине падавина, ваздушног притиска, сунчеве радијације, температуре и влажности земљишта, *IR* радијације, као и времена од изласка до заласка сунца [27]. Одабрани температурни сензор вршио је мерења температуре у опсегу од -60^{o}C do +60^{o}C са тачношћу од ±0.1^{o}C. Тачност сензора за мерење влажности ваздуха износила је ± 5% у опсегу мерања од 0% do 100%. Сензор за брзину и смер ветра покривао је свих 360 степени и имао могућност мерења брзине ветра у распону од 0 m/s до 75 m/s. Грешка сензора за количину падавина износила је 3%, док је грешка при мерењу атмосферског притиска износила ±0,1 hPa при мерењу у опсегу од 920 до 1080 hPa. Сви поменути сензори су били повезани на *CR10X data loger* путем дигиталних и аналогних улазних пинова. Тачније, сензор за количину падавина је био повезан путем дигиталних улазних пинова, док су остали сензори били повезани аналогно. Као комуникациони интерфејс искоришћен је *SC93A CSI data loger* који се може повезати са било којом модемом, конфигурисаним као *RS232 DCE* серијски порт. У овом случају је

искоришћен *M2M Wavecom* модем. Овај модем са *dual band* антеном подржавао је и удаљени приступ и контролу над станицом путем *GSM/GPRS* података, *SMS*-а и гласовних команди путем серијске конекције. Овако конципирана метеоролошка станица је била повезана на константни извор напајања са градске мреже. Постојала је могућност повезивања на напајање са соларног панела или ветрогенератора, а који би вршили пуњење батерије од 12V9 Ah. Имплементирани софтвер је био заснован на клијент сервер архитектури у циљу дистрибуције података између *data logera* и централног рачунара. Флуктрације у измереним вредностима између овако креиране метеоролошке станице и комерцијалне метеоролошке станице биле су занемарљиве, тачније евидентирана су значајна поклапања. Примера ради, коефицент корелације за температуру ваздуха износио је λ=0.999013. Поред значајних предности, евидентирани су и одређени недостаци на чијем решавању би требало радити. Евидентирани недостаци креиране и комерцијалних метеоролошких станица се огледају у немогућности евидентирања мале количине падавина, на пример, количине падавина од 0,05 mm. Аутори предлажу надоградњу постојећих метеоролошких станица камерама које би попут људског ока евидентирале чак и овако малу количину падавина, коју иначе сензори за мерење количине падавина не могу евидентирати.

Систем за мерење различитих метеоролошких параметара дизајниран је у оквиру истраживања описаног у [28]. Овај систем карактерише портабилност, удаљени приступ, као и ниска цена имплементације. Уређај за мерење метеоролошких параметара креиран је као самостална јединица са које су подаци прослеђивани путем *GSM* мреже до станице за обраду истих. За ову намену *GSM* 03,38 протокол је коришћен како би се креирали *SMS* пакети. У склопу овог протокола *timestamp* се састојао од године, месеца, дана, сата, минута, секунде и временске зоне што одговара временској компоненти послатог пакета. Целокупна временска компонента износила је 7 бајтова. Вредности измерених параметра прослеђивани су као интегер вредности и сваки параметар заузимао је 2 бајта. Напајање система врешено је путем батерије, што је узроковало имплементирање додатних механизама за очување енергије у батерији како би модул са сензорима имао константно напајање. За мерење температуре ваздуха коришћен је *LM35*

сензор. Релативна влажност ваздуха мерена је на сваких сат времена коришћењем *HSM 20G* сензора. Тачност овог сензора износи ±5%. Уређај за мерење брзине ветра састојао се од тахометра и ротирајуће елисе. Окретањем елисе врши се усмравање светлости на фотодиоде која се налазила унутар тахометра. Са сваким падом светлости отпор на фотодиоди постаје нула. Помоћу фотодиоде вршено је и мерење јачине светлости.

Креирање економски повољног, флексибилног, портабилног, скалабилног и лаког за коришћење система за мерење метеоролошких параметара био је циљ истраживања описаног у [29]. Систем је креиран како би вршио мерење температуре ваздуха, влажности ваздуха, атмосферског притиска, брзине ветра, количине падавина, присуства росе, *UV* индекса, присуства прашине у ваздуху, јачине амбијенталне светлости, као и присуства различитих гасова у ваздуху. За мерење тренутних температурних вредности, као и у неким од раније поменутих истраживања коришћен је *LM35* темературни сензор. Овај сензор био је повезан са микроконтролером који је врши обраду добијених мерења. Такође, као и у истраживању [28] вредности влажности ваздуха мерене су помоћу *HSM-20G* сензора такође повезаног са микроконтролером. За потребе мерења атмосферског притиска аутори су користили *MPL115A1* сензор. Овај дигитални сензор врши конвертовање података измерених помоћу *Piezoresitive* полупроводничког материјала у дигитални облик. Са повећањем атмосферског притиска, услед механичког стреса, отпорност оваквих материјала се повећава, што се може детектовати и конвертовати у адекватну вредност атмосферског притиска. Према спецификацији произвођача тачност измерене вредности атмосферског притиска је до 1 kPa, док је мерни опсег од 50 kPa до 115 kPa. Поменути сензор поред везе са микроконтролером у исто време био је повезан и са *LCD* дисплејем на коме је вршено приказивање тренутних вредности. Уређаји за мерење брзине ветра и количине падавина креирани су од локално доступних материјала. За мерење брзине ветра коришћен је анометар са полулоптама, због своје линеарности са брзином ветра. Приликом сваке ротације тела анометра, услед дејства ветра, магнет прелази преко сензора за *Hall Effect*. Овај сензор креира прелазни напонски ниво. Прекид на микроконтролеру је коришћен како би се оваква промена детектовала. Бројач у микроконтролеру је коришћен како би се вршило

бројање ротација на сваке 4 секунде. Овакав начин израчунавања коришћен је како би се израчунао број ротација у минути, што је даље еквивалентно брзини ветра. За мерење количине падавина искоришћен је концепт посуде која се сама празни. У циљу тачнијег израчунавања количина падавина по јединици површине искоришћен је левак којим је кишница довођена до боце која је била повезана са механизмом за бројање заснованим на магнету и *A6851 hall effect* сензору. Као резултат са сваким пражњењем посуде овај сензор генерише сигнал од 5 V у главно коло који се региструје од стране микроконтролера. Мерењем сваког сигнала систем приказује укупну количину падавина у mm за одређени период времена. За мерење *UV* индекса *GUVA-S12SD* фотодиода је коришћења. Са повећањем *UV* индекса излазни напон фотодиоде се такође повећава, што је у исто време искоришћено као законитост за израчунавање вредности *UV* индекса. Мерење интензитета амбијенталне светлости вршено је помоћу Temt6000 фототранзистора високе осетљивости на видљиво светло. Фототранзистор је био повезан са микроконтролером у циљу мерења излазног напона и израчунавања одговарајуће амбијенталне светлости. Однос ове две величине такав је да се са повећањем интензитета амбијенталне светлости повећава и напон на микроконтролеру. Оптички сензор *GP2Y1010AU* је коришћен за мерење густине прашине у ваздуху. Сензор се састоји од инфрацрвене диоде и фототранзистора који су постављени дијагонало, тако да су усмерени једно ка другом. Сензор ради тако што детектује количину светлости која се рефлектује од прашине у ваздуху. Излазни напон са сензора, доведен на микроконтролер, представља количину прашине у ваздуху. Зависност између излазног напона и количине прашине у ваздуху је линеарна. За мерење концентрације различитих гасова у ваздуху коришћени су сензори *MQ4*, *MQ7* и *MQ135*. Сензор *MQ4* је коришћен за детектовање природног гаса у ваздуху, *MQ7* за детектовање угљен-моноксида, док је *MQ135* коришћен за детекцију присуства угљен-диоксида, дима и амонијака у ваздуху. Као и претходни сензори и овај сет сензора повезан је са микроконтролером. За потребе слања измерених вредности параметара до сервера искоришћен је *SIM900 GSM* модул. Слање података вршено је путем *SMS*-а и *GPRS*, како до корисника са унапред унесеним бројевима телефона тако и до web сервера респективно. Корисници су могли да прате тренутне вредности свих

метеоролошких параметара покретањем *PHP* web стране која са датог сервера преузима податке из *SQL* базе података. Напајање система обезеђено је путем соларног панела и батерије, што га чини лако портабилним и енергетски независним. За активан рад целокупног система током 24 сати потребно је 36 Wh енергије. Укупна доступна енергија овако креираног система је 84 Wh. С обзиром да је стални рад *GSM* модула јако енергетски захтеван, овај модул је од стране микроконтролера активиран у тачно предвиђено време и изнова враћан у неактиван мод одмах након успешног слања података. Поређењем вредности добијених са овако креиране метеоролошке станице, са вредностима добијеним са комерцијалне метеоролошке станице, аутори су закључили да није било значајних одступања у мерним вредностима. Поред великог степена тачности, још једна од кључних погодности је и цена целокупног система која износи око 240 еура. У поређењу са комерцијалним метеоролошким станицама економски издаци овако креираног система могу се сматрати занемарљивим, што га чини доступним за мала подручја и пољопривредне површине.

Систем за праћење метеоролошких параметара и стања атмосфере у једном од истраживања био је заснован на две платформе [30]. Прва платформа представљала је мерну станицу чији је рад заснован на микроконтролеру, *GPS* систему, *GSM* модулу и сензорима за праћење атмосферских појава. Друга платформа је представљала станицу за обраду података и заснована је на лап топ или десктоп рачунару са *GSM* модемом, у циљу адекватног бежичног повезивања са мерном станицом. Сензор *BMP085* искоишћен је за мерење атмосферског притиска и температуре, док је *SY-HS-220* сензор искоришћен за мерење релативне влажности ваздуха. Поменути сензори су били повезани са LPC2148 микроконтролером. Такође, аутори су са микроконтролером повезали и GMS *SIM900* модул, као и *GPS* модул. *GPS* модул је имплементиран за потребе слања информација о позицији мерне станице (географска ширина, георафска дужина и надморска висина), као и времену слања података. Тачност температурног сензора износила је ±0,1^{o}C, тачност сензора за мерење атмосферског притиска износила је 0,01 hPa, док је тачност сензора за мерење релативне влажности ваздуха износила 1%. Мерни опсег имплементираних сензора износио је од 30%

до 90%, од 300 hPa до 1.100 hPa и од -40^{o}C до +85^{o}C, за релативну влажност ваздуха, атмосферски притисак и температуру, респективно.

Метеоролошки услови један су од веома значајних параметара у развоју биљних патогена. Практично поред периода у току године, у коме може доћи до појаве одређеног биљног патогена, и метеоролошки услови за развој истог морају бити задовољени. Прецизније, присуство патогена је потребан али не и довољан услов за инфекцију. Метеоролошки услови су ти који за сваки од патогена престављају довољан услов за успешан развој и инфекцију биљке. Праћење метеоролошких услова и њихове повезаности са развојем биљних патогена веома је значајно, па је самим тим и предмет великог броја истраживања. У једном од истраживања метеоролошки подаци прикупљени са седам локалитата у Аустралији, Бразилу и Колумбији коришћени су за креирање модела за предикцију могућности појаве антракозе на ливадском биљу [31]. Подаци о појави антракозе и метеоролошки подаци анализирани су помоћу вештачких неуронских мрежа. На основу података из прошлости, забележених на територији Аустралије и Јужне Америке, креирани су предикциони модели како би се на бази нових података извршила предикција појаве ове болести на истим или другим регионима. Локације на којима су прикупљани тренинг и тест подаци биле су географски удаљене, с обзиром са су се налазиле на различитим континетима или у различитим државама. Ови модели су креирани на бази различитих метеоролшких података (минимална и максимална дневна влажност ваздуха, минимална и максимална темепратура ваздуха, укупно време радијације, дужина трајања влажности листа, укупно време трајања ветра, количина падавина и број сунчаних сати). Вредности метеоролошких параметара прикупљане су са метеоролошке станице која је била опремљена сензорима за мерење темературе ваздуха, релативне влажности ваздуха, влажности листа, као и сензором за мерење трајања ветра. Сензори за мерење количине падавина и влажности листа вршили су мерења на сваких 6 минута, од тренутка првог евидентирања промена. Поред предикције могућности појаве болести, циљ је био установити који метеоролошки параметри највише утичу на појаву антракозе. Највећа стопа успеха забележена је приликом рада са метеоролошким подацима прикупљеним на дан контроле присиства антракозе на пољу, као и са подацима из претходних

24 h. У више од 75% посмараних дана, на основу метеоролошких података са једног континента, аутори су помоћу креираних модела успели да изврше предикцију појаве болести на другом контитенту. Предикциона грешка за Аустралију износила је 21,9%, док је за Јужну Америку износила 22,1%. Креирани предикциони модели дали су тачност предикције у распону од 54% до 96%. На основу multiple regression модела установљено је да је влажност један од најбитних параметара за развој овог патогена. Такође, регресиони модели су показали да и метеоролошки параметри, као што су киша, влажност листа, ветар, зрачење, утичу на појаву болести самим тим што доводе до довољног степена влажности.

Систем за одређивање присуства патогена на листовима биљака, као и типа биљке којој лист припада заснован на *Back Propagation* неуронским мрежама креиран је са циљем адекватне контроле и идентификације заражених биљака [32]. Креирани софтверски систем састојао се од пет модула. У првом од модула, под називом *Leaves Processing,* скенирана слика листа се обрађује тако што се одређују контуре листа и маркирају места на листу која могу представљати симптоме напада патогена. У модулу под називом *Network Training* вршено је тренирање целе неуронске мреже и исцртавање графа грешке. Трећи у низу модула, под називом *Recognition Module,* био је задужен за препознавање листа и одређивање процентуалног поклапања са листовима у бази података на основу којих је неуронска мрежа обучена. Четврти модул *Pest Recognition* коришћен је за одређивање процентуалне заступљености симптома болести на листу. Последњи модул је служио за проналажење поклапања између информација добијених са новоскенираног листа и података сачуваних у бази података који представљају већ креиране патерне. Такође, у овом модулу корисник је добијао повратну информацију о поступцима које је потребно применити ради ефикасне заштите. Сви подаци о поменутим поступцима раније су сачувани у бази података. На основу детектованих маркера на површини скенираног листа креирани су улази у *Feed-Forward Back Propagation* неуронску мрежу. Улаз у неуронску мрежу су индивидуални токени слике листа. С обзиром да се токен уобичајено састоји од косинуса и синуса угла, број улазних слојева ове мреже једнак је броју токена пута два. Синус и косинус угла облика заправо представљају критеријум за

одређивање патерна. Број излазних неурона обично је одређен бројем различитих врста због коришћења кодираног облика приликом специфицирања излаза. У тренинг фази *Back Propagation* алгоритам позиван је тачан број пута над отвореним фајлом са улазним подацима. Након завршетка алгоритма, коначне добијене тежине су чуване у излазном фајлу који је касније коришћен у фази препознавања патерна.

Често није једноставно пронаћи све податке који су потребни ради креирања модела и успешне предикције. У таквим случајевима одређени подаци се могу добити предикцијом на основу података из прошлости, како би се даље користили у новим предикционим моделима. Метеоролошки подаци се могу, на основу оних из прошлости, добити предикцијом за предстојећи период, а онда користити у предикционим моделима за предикцију болести. Није редак случај да се моменат или начин напада добију предикцијом, а затим даље учествују у новим предикцијама. Нека од истраживања заснована на оваквим методама описана су у наставку.

Комбиновањем метеоролошких и биолошких параметара у једном од истраживања креиран је фази логички систем за процену тренутне стопе инфекције и појаве рђе на соји [33]. Самим тим, аутори су креирали фази логички модел који врши симулацију присуства рђе на соји на основу података добијених са експерименталних поља истраживачког центра на Тајвану. Овај модел врши процену дневне стопе инфекције, као и симулацију озбиљности напада на основу популационе динамике. Симулацијом целокупног периода у коме може доћи до епидемије, модел је успешно вршио предикцију у случајевима када су иницијалне вредности о јачини напада и метеоролошким условима тачно предвиђене. Ово праткично значи да креирани модел може на основу метеоролошких података добијених предикцијом успешно извршити предикцију појаве сојине рђе, као и јачине напада пре почетка периода у коме може доћи до инфекције. Базични метеоролошки подаци потребни за развој модела били су температура ваздуха, период росе и количина падавина. Ови подаци су добијени мерењем помоћу метеорлошке станице, док су очитавања вредности вршена на свака два сата. Подаци о појави болести, добијени из експеримената спроведених у периоду од

1980. до 1981. године, искоришћени су за равој модела и његову евалуацију, респективно. Датум почетка инфекције постављен је 14 дана након садње под претпоставком да је инфекција почела чим је соја развила прве праве листове. Почевши од дана када је појава болести уочена јачина напада одређивана је симулацијом за седмодневни период помоћу креираног модела који врши процену дневне стопе напада. Добијене вредности су упоређиване са вредностима добијеним мерењима у циљу одређивања тачности процењене дневне стопе напада. Овакав алат може дати добру подршку, како за одређивање потенцијалне јачине напада тако и за одређивање временског периода у коме је најадекватније обавити заштиту фунгицидима.

Предикциони модели базирани на класификационим и регресионим стаблима одлучивања (*CART* модел) и фази логички (*FL* модел) коришћени су у циљу предикције дужине трајања влажности листа, 24h унапред. Овако добијени податак коришћен је као улаз за симулацију перформанси *Melcast* и *TOM-CAST* система за упозорење на оствареност услова за појаву биљних болести. Предикција влажности листа базирана је на подацима прикупљеним са 15 стандардних метеоролошких станица у Ајови, Илионсу и Небраски, у периоду од маја месеца до септембра месеца 1998. и 1999. године. Метеоролошки параметри на којима се базирала предикција су: температура ваздуха, релативна влажност ваздуха и брзина ветра. Влажност ваздуха је мерена помоћу електронског сензора (*Model 237; Campbell Scien- tific, Logan, UT*). Уколико је влажност детектована 30 и више минута у току једног сата тај сат је евидентиран као 1, уколико није сат је евдентиран са 0. Предикциона грешка износила је 2,3 и 3,9 часова дневно за *CART* и *FL* модел респективно. Оба модела у 45% случајева успешно су извршила предикцију дана у коме је потребно врштити заштиту биљака. Резултати истраживања су показали да тачност овако креираног модела мора бити унапређена у циљу боље предикције момента за адекватну хемијску заштиту [34].

Систем за предикцију појаве болести и штеточна на воћу у округу *Qixia* у Народној Републици Кини заснован на метеорлошким параметрима реализован је у једном од истраживања применом *Back Propagation* неуронских мрежа. За потребе тренирања неуронске мреже и имплементације система истраживачи су

користили *MatLab neural netvork toolbox* [35]. Метеоролошки подаци и подаци о појави болести за период од 1992. до 2000. године коришћени су као сирови подаци на основу којих је вршено тренирање предикционог модела. Скуп података за период од 2001. до 2002. године коришћен је у процесу валидације предикционих резултата. У оба случаја метеоролошки подаци (просечна дневна температура, количина падавина, број кишних дана, влажност ваздуха, брзина и смер ветра) сваке године евидентирани су у периоду од априла до августа. Такође, за сваку од година појава болести и штеточина у периоду од маја до септембра добијена је у сарадњи са људима из области заштите биља, на основу њихових података са терена. Поред метеоролошких података и података о појави болести и штеточина, информација о месецу у коме су подаци прикупљени коришћена је као један од улазних параметара предикционог модела. У циљу ефикаснијег и бржег тренирања неуронске мреже сви подаци су нормализовани. У вези са појавом болести и штеточина креирана је скала нормализације 0.1, 0.3, 0.5, 0.7 и 0.9, што одговара текстуалном опису јачине напада: лоше, мање лоше, средње, лакше, најлакше респективно. Креирана *Back Propagation* неуронска мрежа садржала је три слоја (улазни, скривени и излазни). Нелинеарно мапирање од улаза до излаза било је засновано на матрици тежина. Матрица тежина креирана је од улаза до скривеног слоја, и од скривеног слоја до излазног слоја. Након тренинг процеса, тежинске вредности чворова у мрежи су ажуриране у зависности од степена грешке између добијене и очекиване вредности на излазу. Како је *Back Propagation* алгоритам заснован на *gradient descent*-у основна идеја је измена вредности тежина у циљу минимизације грешке суме квадрата (енг. *sum-square*) између очекиване и добијене вредности излаза. Број чворова у скривеном слоју је повезан са комплексношћу проблема и има директног утицаја на особину нелинеарности мреже. Повећањем броја чворова у скривеном слоју, повећава се и тачност тренинг процеса. Тачност предикционог модела добијена је поређењем вредности добијених предикцијом и стварних вредности о појави конкретних болести. Стопа грешке износила је 7.14% и 9.29% за 2001. и 2002. годину, респективно. Овакви резултати показују да је метода предикције појаве болести и штеточина помоћу неуронских мрежа изузетно поуздана.

Различити модели за предикцију појаве пегавости листа озиме пшенице базирани на 431 регистрованом случају ове болести у периоду од 2012. до 2014. године тесирани су у истраживању групе аутора [36]. Током 2012., 2013., и 2014. године регистровано је 35, 236 и 160 случајева, респективно. Сви регистровани случајеви са јединственим комбинацијама предиктора били су рандом подељени у тренинг (70% целокупног скупа података), валидационе (20% целокупног скупа података) и тест (10% целокупног скупа података) скупове података. Креирани скупови података коришћени су у циљу креирања једног *MR* (енг. *Multiple Regression)* предикциног модела, као и три модела машинског учења (енг. *Artificial Neural Networks-ANN, Categorical And Regression Trees-CART* и *Random Forests-RF*). Задатак ових модела била је предикција ризика појаве пегавости листа различитих сорти озиме пшенице. Одабране независне предикционе променљиве на којима је базирана предикција биле су отпорност сорте пшенице, географска ширина и географска дужина парцеле на којој се налази пшеница, претходна култура, густина сејања, третман семена, тип обраде и стопа остатака пшенице на производној површини из претходних година. Вредности ових променљивих добијене су са производних површина лоцираних у 12 градова који су се налазили у 11 земаља Северне Каролине. Све независне предикционе променљиве су нормализоване, или уколико је то могуће, за исте је уведено кодирање нулом или јединицом у циљу адекватнијег креирања предикционог модела. Максимална јачина регистрованог напада на крају производне сезоне коришћења је као излазна променљива предикционог модела. Ова вредност добијена је визуелним прегледом производних површина неколико пута током вегетационе сезоне. Модели су евалуирани на основу разлике између вредности јачине напада добијене предикцијом и стварне вредности јачине напада регистроване на производним површинама. У процесу евалуације модела коришћена је *ROC* анализа и *Kappa* статистика. Евалуација модела је показала да су географска дужина, географска ширина, отпорност сорте и остаци пшенице независне предикционе променљиве од којих у највећој мери зависи тачност предикције. Евалуациони резултати показали су да је *MR* модел успешно класификовао 74% случајева појаве болести, док је тачност класификације алгоритама машинског учења износила између 81% и 83%. Такође, резултати

показују да је тачност *RF* алгоритма машинског учења била највећа и износила је 93%. Закључак аутора је да се предност коришћења овог алгоритма огледа се у раној процени ризика појаве ове болести, што у многоме може олакшати доношење одлуке о благовременој заштити, чак и пре сетве пшенице.

Различите предикционе технике (*Multiple regression, Back-propagation neural network, Generalized regression neural network, Support vector machine, Support vector regression*) коришћене су у циљу креирања предикционог модела у циљу предикције појаве болести на листу пиринча [37]. Предикција је била базирана на метеоролошким параметрима и подацима о јачини напада регистрованим током 2000. године, као и на подацима добијеним у склопу екперименталног пројекта који је трајао у периоду од 2001. до 2004 године. Метеоролошки подаци (максимална и минимална температура, максимална и минимална релативна влажност ваздуха, количина падавина, као и број кишних дана) на недељном нивоу прикупљани су са аутоматске метеоролошке станице. Средње вредности метеоролошких параметара на недељном нивоу су израчунаване како би се користиле као независне променљиве у процесу предикције појаве болести. *SPSS Statistics* софтвер је коришћен у поступку крерања, обуке и валидације *Multiple regression* предикционог модела. *Back Propagation neural network* модел базиран на поменутим параметрима креиран је и валидиран помоћу *Stuttgart Neural Network* симулатора, док је *Generalized regression neural network* модел креиран у *MatLab*-у. *SVM_light* имплементација у *C*-у коришћена је за креирање *Support Vector Machine* и *Support Vector Regression* предикционих модела. *Five-fold* унакрсно валидациона процедура базирана на локалитетима, као и *five-fold* унакрсно валидациона процедура базирана на годишњем скупу података, коришћене су у процесу тестирања свих креираних предикционих модела. Локациона валидација је рађена над подацима који представљају целу календарску годину прикупљени на пет локалитета при чему су четири локалитета коришћена као тренинг скуп док је пети локалитет коришћен као тест скуп података. Слична процедура је коришћена у приликом валидације вишегодишних модела тако што је издвајана по једна година за тест скуп док су подаци који представљају преостале четири године коришћени као тренинг скуп. Аутори су тачност свих креираних предикционих модела одредили на основу

коефицијента детерминације и вредности средње апсолутне грешке у односу на стварне вредности. Истраживање је показало да *Support vector machine* предикционе технике показују бољу тачност него што је то случај са осталим имплементираним моделима.

Предикциони модели базирани на data mining алгоритмима креирани су са циљем предикције појаве болести и штеточина на листу кивија. Истраживање је спроведено од стране групе аутора са Новог Зеланда. Скуп од 80 независних атрибута подељен у три категорије, на бази којих је вршен тренинг предикционих модела, креиран је на основу евиденције хемијских третмана и мониторинга штеточина. Тренинг скуп података креиран је током 2008. и 2009. године, док су у процесу евалуације предикционих модела коришћени подаци из 2010. године. Креирани модели коришћени су у процесу доношења одлуке током производне 2011. године. У процесу креирања предикционих модела коришћена су пет алгоритма машинског учења (*Decision Tree, Naive Bayes, Random Forest, AdaBoost* и *Support Vector Machine*), као и један статистички метод (*Logistic regression*). Сви алгоритми се сврставају у групу класификатора, код којих је циљ био доношење одлуке о моменту примене инсектицида на основу података о томе да ли је популација инсеката изнад или испод унапред дефинисаног прага хемијске заштите. Доношење одлуке о томе да ли ће се хемијски третман вршити или не, базирано је на унапред дефинисаном степену предикционе тачности (процентулано поклапање са коректно класификованим инстанцама у тест резултатима). Евалуација предикционих резултата је показала да је тешко са значајним степеном тачности одредити да ли је потребно вршити хемијски третман. Са друге стране, применом поменутих алгоритама над овако дефинисаним скупом података могуће је са знатно већим степеном тачности донети одлуку да нема потребе за хемијским третманом. Истраживање је показало да је у 49% блокова података тачност доношења одлуке о неспоровођењу третмана применом *AdaBoost* алгоритма износила 98%, док је у 70% блокова података тачност *Naive Bayes* алгоритма износила 95% [38]. Анализа података, отклањање поклапања, обука модела и предикција су вршени у *Weka data mining workbench*-у кроз *ADAMS* (енг. *Advanced Data mining and Machine learning System*) окружење.

Заједничко истраживање спроведено од стране истраживача из Индије и Јапана имало је за циљ креирање система за доношење одлука у реалном времену који је назван *GeoSense* и који је продукт интеграције *Geo-ICT* и бежичне сензорне мреже [39]. Примена овог система оријентисана је ка примени у прецизној пољопривреди. Практично систем је развијан како би се утврдила корелација између биљака и штеточина, као и корелација између појаве болести и метеоролошких услова. Метеоролошки подаци (температура ваздуха, влажност ваздуха и влажност листа) прикупљени помоћу бежичне сензорне мреже, као и подаци о појави штеточина и болести на биљкама, добијени посматрањем производних засада, анализирани су помоћу data mining техника. Сви подаци на којима је базирана корелација појаве болести и популација штеточина прикупљени су током четири производне сезоне. Додатно су креирани предикциони модели за предикцију појаве штеточине *Thrips* и *BNV* (енг. *Bud necrosis* virusa) који ова штеточина преноси на површу. Систем за прикупљање метеоролошких података састојао се од мерних чворова који су се напајали помоћу батерије и вршили слање пакета података од једног до другог чвора на сваких 15 минута. Максимално растојање између два чвора у мрежи је било 25 m. Пренос података вршен је од чвора до чвора све док се подаци не доставе до базне станице. Базна станица је била опремљена GPRS конекцијом помоћу које је слала податке до *FTP* сервера, а који је вршио складиштење података. Подаци о популацији *Thrips-a* и појави *BNV* вируса су добијени посматрањем производних засада сваке недеље од почетка цветања па све до фазе репродукције. Овај период одабран је јер се већина штеточина и болести јавља у периоду између ове две фенофазе развоја биљке. *Gaussian Naive Bayes* data mining алгоритам је коришћен у процесу класификације података, *Rapid Association Rule* data mining алгоритам је коришћен за асоцијациону и корелациону анализу података, док је *Expectation–Maximization* алгоритам коришћен у процесу одређивања вредности података који недостају у инстанцама. *Regression* data mining технике су коришћене у тренинг процесу предикционих модела, након чега је вршена предикција. Сви поменути алгоритми примењени су над подацима у оквиру *Weka data mining* алата. Све *multivariate regression* једначине које дају везу између појаве *Thrips-a*, метеоролошких параметара и старости биљке генерисане су помоћу *XLminer Add-*

in data mining алата u *Excel*-u. Након тога, индекс инфекције *BNV* вирусом израчунаван је на основу предиктивне вредности популације *Thrips*-а помоћу креираних емпиријских модела. Креирани емпиријски предикциони модели су као резултат дали везу између критичне популације ове штеточине и вируса који она преноси на биљке. Овакав систем у многоме ће олакшати доношење одлука у циљу заштите биљака од штеточина и болести.

4. Методе истраживања

Велики скупови података се обрађују различитим техникама, чији је крајњи циљ откривање корисног знања. Ове технике припадају процесу који се назива откривање знања у базама података (енг. *Knowledge Discovery in Databases-KDD*). Data mining је кључни основ у *KDD* процесу, који укључује коришћење алгоритама за истраживање података, развијање модела и откривање претходно непознатиих шаблона и веза између података. На овакав начин креирани модели се користе у циљу утврђивања међусобне зависности између података, анализе података и предикције. Data mining се не може посматрати као колекција изолованих података од којих је сваки потпуно другачији од другог и који чекају да се примене на дати проблем. Уколико се посматра процес обраде података, може се уочити да data mining није коришћење рандом аналитичких техника, већ је пажљиво планиран и разматран процес одлучивања о томе шта је најкорисније и најрелеватније применити како би се задати циљ остварио. Према таксономији data mining-а могу се разликовати два главна типа: верификација и откривање. Верификација представља методе којима се врши потврђивање хипотезе, док откривање представља самостални систем који проналази нова правила и шаблоне између податка. Два већ поменута главна задатка data mining-а, предикција и дескрипција, припадају скупу метода откривања [5]. Целокупна таксономија data mining-а дата је на слици 1.

Слика 1: Таксономија Data mining-а

Већина data mining техника, које припадају групи откривања, базирају се на индуктивном учењу, где је модел креиран експлицитно или имплицитно генерализацијом из довољног броја тренинг примера. Терминологија која је у употреби од стране заједнице машинског учења предикционе методе data mining-а сврстава се у категорију надгледаног учења (енг. *Supervised Learning*). Као супротност надгледаном учењу издваја категорију ненадгледаног учења (енг. *Unsupervised Learning*). Ненадгледано учење се најчешће односи на технике које групишу инстанце података без унапред дефинисаног зависног атрубута. Ненадгледано учење представља приступ проблему учења који се односи на ситуације у којима се алгоритму који учи пружају само подаци без излаза, а од алгоритма који учи очекује се да сам уочи неке законитости у подацима који су му дати. Коришћење ненадгледаног учења представља прави изазов у примени над проблемима проналажења правог алгоритма. Веома је тешко одредити да ли је задатак урађен како треба или није, што је последица непостојања одговарауће метрике. Управо због недостатка метрике, као и непостојања класног атрибута код ненадгледаног учења скоро је немогуће добити објективну меру о тачности примењеног алгоритма. На пример, од примене *K-means* алгоритма кластеризације не постоји механизам којим би се одредило да ли је овај алгоритам креирао одговарајуће кластере, да ли је број кластера дефинисан на почетку процеса био одговарајући итд. Један од начина за тестирање креираног модела ненадгледаног учења је његова имплементација и праћење у стварном свету. На овакав начин може се утврдити да ли излаз алгоритма даје тачне или нетачне информације. Посматрајући таксономију data mining-а ненадгледано учење би обухватало део дескриптивних метода. Ненадгледано учење обухвата кластеризационе технике, али не и методе визуализације података.

Ненадгледано учење може се класификовати у две категорије:

- параметризовано ненадгледано учење;
- непараметризовано ненадгледано учење.

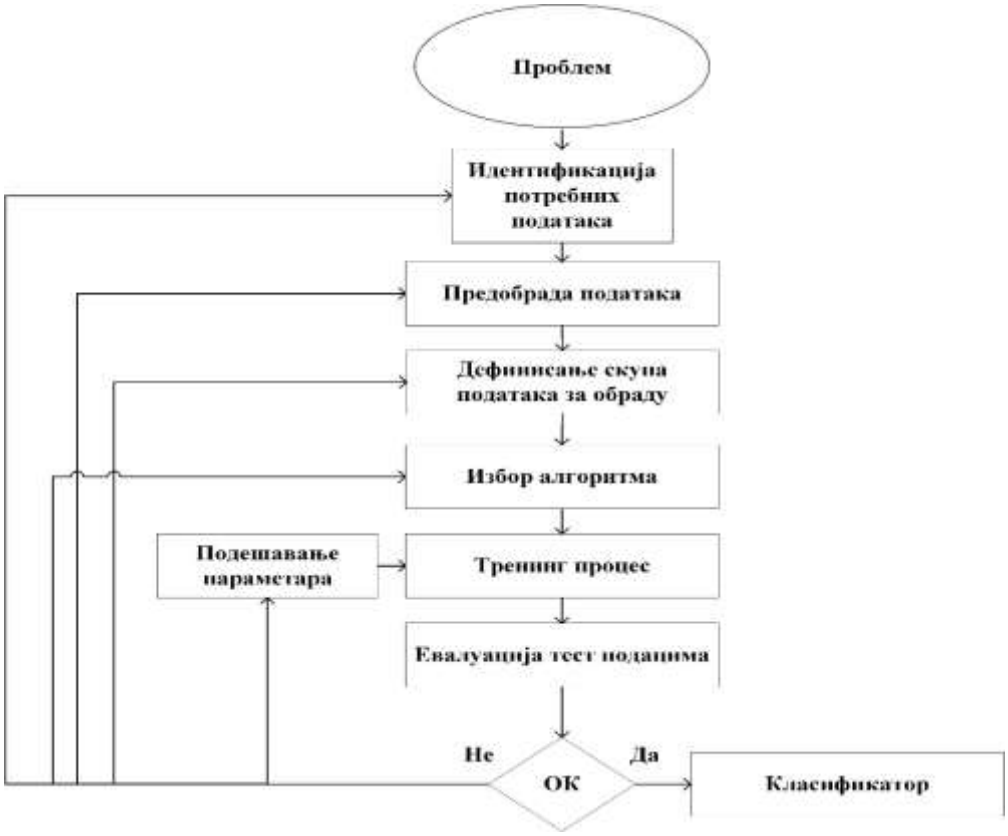
Под параметризованим ненадгледаним учењем подразумева се параметризована дистрибуција података. Претпоставља се да је скуп података који се користи као узорак део веће популације која прати расподелу вероватноће

засновану на фиксном скупу параметара. Теоретски у нормалној дистрибуцији сви чланови имају исти облик и параметризовани су на основу средње вредности и стандардне девијације. Применом на будуће одлучивање, уколико је позната аритметичка средина и стандардна девијација, као и податак да је упитању нормална дистрибуција, онда се лако може одредити вероватноћа будућег посматрања. Параметризовано ненадгледано учење укључује конструкцију Гаусовог модела и коришћење *Expectation-Maximization* алгоритма у циљу предикције класе узорка који се посматра.

У непараметризованом ненадгледаном учењу подаци су груписани у кластере. Поступком груписања података у кластере требало би да се добије више информација о категоријама и класама података присутним у самом скупу података. Овакав поступак најчешће се користи код моделовања и анализе мањих скупова поатака. За разлику од парамеризованих модела, непараметризовани модели не захтевају увођење било каквих претпоставки о дистрибуцији података у популацији, те се из тог разлога посматрају и као методи независни од дистрибуције.

Методе надгледаног учења отривају везе између улазних атрибута који се још називају и независне променљиве. Уобичајено је да модели креирани методама надгледаног учења описују везе и феномене који су скривени у скупу података и могу се користити, као што је раније наведено, у поступку предикције зависне променљиве. Надгледано учење може бити примењено на различе скупове података. Сам процес практичне реализације надгледаног машинског учења, према [40], на реалне проблеме који нас окружују, дат је на слици 2.

Код надгледаног учења, за разлику од ненадгледаног, постоји дефинисана метрика која се користи у процесу доношења одлуке или предикције. Метрике као што су прецизност и *Recall* дају довољан податак о томе колико је креирани модел тачан. Такође, параметри модела су подешени тако да повећају тачност самог модела. Мала тачност указује да је потребно унапредити сами модел. Два главна модела надгледаног учења су класификациони и регресиони модели.



Слика 2: Процес надгледаног учења

Према слици 2 може се уочити да је крајњи продукт надгледаног учења на реалне проблеме класификатор, што значи да дата шема представља процес надгледаног учења применом класификационих модела. Примена регресионог модела би као крајњи продукт дала регресиони модел. Процес реализације креирања регресионог модела је практично исти као и креирање класификационог модела (класификатора). Поред сличног поступка реализације између два основна модела надгледаног учења могу се уочити и одређене разлике. Применом регресионих техника може предвидети потражња неког производа на основу његових карактеристика. Са друге стране, класификационим техникама могу се мапирати улазни подаци у предефинисане класе, те се може вршити предикција која ће на основу истих података имати различит излаз, те се тако може бирати између већег број класних атрибута.

Независно од тога да ли се ради о надгледаном или ненагледаном учењу, као и о томе да ли се ради о класификационим или регресионим техникама или о техникама за реализацију параметризованог или непараметризованог учења,

процес решавања проблема применом data mining техника састоји се од следећих фаза:

- дефинисање проблема и формулисање хипотезе;
- прикупљање података;
- предпроцесирање продатака;
- одабир одговрајуће data mining технике;
- интерпретирање модела;
- евалуација модела;
- коришћење добијеног знања.

Свака од наведених фаза се може реализовати применом различитих data mining техника. Процес почиње дефинисањем проблема, дефинисањем домена примене data mining техника и дефинисањем очекиваних крајњих циљева обраде података. На самом почетку пројектанти модела морају дефинисати циљеве крајњег корисника и окружење у коме процес откривања знања може заузети одговарајуће место. Током трајања процеса дефинисана почетна хипотеза и крајњи циљеви подложни су ревизији. Друга фаза решавања проблема и обраде података, у циљу реализације задатих циљева, је прикупљање података и њихова организација. Како би се креирао адекватан сет података потребно је најпре извршити селекцију потребних података. Ово укључује утврђивање доступних података, прикупљање додатних потребних података, интеграцију свих података у одговарајуће скупове података и одређивање атрибута на којима ће бити заснован процес доношења одлука или добијања корисног знања. Избор одговарајућих атрибута, односно атрибута који су највише информативни, може бити сугерисан од стране адекватног стручњака, под условом да је на располагању. Овај процес је веома битан због чињенице да се обука и примена креираног модела data mining техникама заснива на доступним подацима, што је основа за конструкцију адекватног модела. Уколико одређени важни атрибути недостају, целокупно истраживање може пропасти. За што бољи успех будућег data mining модела потребно је у овој фази одабрати што је могуће више атрибута. Проблем који се може јавити у овој фази јесте превелик скуп атрибута, а самим тим и превелик скуп података. Складишта података која се користе за прикупљање, организовање

и управљање сложеним подацима веома су скупа, те се из тог разлога мора пронаћи компромис са могућностима за најбоље разумевање феномена. Овај однос почиње најбољим доступним скупом података који се касније шири, при чему се посматра ефекат у домену откривања знања и моделовања. Како би подаци који су прикупљени и од којих је креиран скуп података били што адекватнији могу се дефинисати индикатори квалитета којима се треба водити у току фазе прикупљања података и касније у фази предпроцесирања [6]. Неки од индикатора квалитета могу се дефинисати на начин приказан у наставку:

- Подаци морају бити тачни. Нетачни подаци могу довести до погрешних крајњих резултата. Што се тиче тачности података у зависности од типа података врше се различите провере. На пример, уколико се ради о текстуалним подацима врши се провера да ли су речи написане коректно или уколико се ради о кодовима да ли су кодови у датом опсегу, да ли су нумеричке вредности целобројне или реалне итд.
- Подаци треба да буду смештени према типу података, што значи да нумерички подаци не смеју бити представљени као карактери, цели бројеви као реални итд.
- Подаци треба да имају интегритет. Питање интегритета се односи на ажуриране скупове података. Практично старије верзије се не смеју трајно изгубити како не би дошло до конфликта између различитих корисника. Велики *back*-апови и процедуре враћања на претходно стање могу се имплементирати као део базе података. На овакав начин се осигурава интегритет података, како у почетним фазама тако и у свим наредним.
- Подаци треба да буду доследни. Приликом спајања већег броја скупова података у један скуп података може доћи до најразличитијих проблема. Како би се избегли потенцијални губици података и нарушавање структурне уређености, форма и садржај треба да остану исти након интерације великих скупова података са различитих извора.
- Подаци не треба да буду редундантни. У пракси редундантни подаци треба да буду миниминизовани, док оправдано дуплицирање података треба да буде контролисано или се дупликати записа требају елиминисати.

- Подаци треба да буду благовремени. Временска компонента података треба да буде препозната експлицитно или имплицитно од стране организације.
- Подаци требају да буду разумљиви. Стандарди у називима су потребан, али не и довољан и једини услов за податке како би они били разумљиви.
- Подаци треба да буду комплетни. Подаци који недостају, што се у реалности дешава, морају бити минимизовани. Овакви подаци могу редуковати квалитет глобалног модела који се креира у будућим фазама data mining процеса.

Сет сирових података креиран за потребе обраде data mining техникама у највећем броју случајева је велики, док су подаци у њему најчешће прикупљани посредством човека, самим тим садрже одређену количину шума. Може се очекивати да у скупу података недостају поједине вредности или да су неке вредности погрешно унете. Могу се очекивати дисторзија, несортираност и неадекватни узорци. Сви ови недостаци су својствени човеку, тако је готово немогуће имати сирове податке без оваквих карактеристика. Из тог разлога се примењују технике за реализацију решавања недостајућих података, технике за откривање шума, при чему свака од техника носи са собом одређене предности и недостатке. Подаци могу да недостају из великог броја разлога. Такође се јављају грешке у мерењима или запису вредности, али у већини случајева вредности су недоступне. Неке од data mining техника су мање или више осетљиве на недостатак података [41].

Још један од проблема се јавља када подаци нису из популације из које се сматра да јесу. Типичан пример су изузетци. Они захтевају анализу пре него што се одучи да ли их треба одбацити из самог процеса одлучивања или укључити као нестандардне примере популације која се проучава. Веома је важно да се подаци испитају пре било какве формалне анализе. Са данашњом величином скупова података ово је процес који је у већини случајева скоро немогуће ручно урадити. Из тог разлога посао отклањања недостатака у подацима се поверава рачунарском програму који ће то обавити уместо човека.

Као што је раније речено, подаци требају да буду добро дефинисани, доследни и непроменљиви по природи. Количина података треба да буде довољно велика да

подржи анализу података, извештавање и поређење историјских података током дугог периода времена. Једна од најкритичнијих фаза у data mining процесу је предпроцесрање у оквиру кога се могу издвојити трансформација и припрема иницијалног сета података као два главна задатка. Процес трансформације података има за задатак смањење величине скупа података и обраду података у оквиру скупа, у циљу његовог довођења на меру погодну за даљу обраду у току фазе припреме податка. Процес припреме података се своди на организовање података у стандардну форму која је спремна за обраду од стране data mining техникаи на то да се скуп података треба припремити тако да води најбољим data mining перформансама.

4.1 Трансформација података

Често припрема сирових података за процес учења се узима само као једна од фаза у data mining процесу. Такође, често се у литератури процес припреме занемарује. Приликом примене data mining техника у реалним ситуацијама и раду са великим скуповима података ситуација је обрнута, што значи да се много више напора улаже у току процеса припреме података него у примени других data mining метода. Процес одабира технике која ће се применити у циљу трансформације података зависи од типа податка у скупу података, количини података у скупу и од генералних карактеристика data mining задатка [6]. Основни типови трансформација које се могу применити над подацима, које у исто време нису зависне од проблема који подаци описују и које воде унапређењу резултата примене data mining техника су:

- нормализација (енг. *Normalizations*);
- уједначавање података (енг. *Data Smoothing*);
- Разлике и односи међу подацима (енг. *Differences and Ratios*).

4.1.1 Нормализација

Неке од data mining метода, типично оне које се базирају на израчунавању дистанце између тачака у *n*-димензионалном простору, имају потребу да подаци над којима се примењују буду нормализовани како би се добили најбољи могући резултати обраде. Овим поступком вредности података у скупу сирових података скалирају се у неком специфичном опсегу, на пример ,−1,1. или ,0,1.

Уколико вредности нису нормализоване, мера растојања између тачака података ће бити већа у односу на оне карактеристике које у просеку имају већу вредност. Постоји више начина за нормализацију података. Примери ефикасних начина нормализације који се не карактеришу великом сложеношћу описани су у наставку.

Децимално скалирање, као један од начина нормализације, врши уклањање децималне тачке у подацима чувајући оригиналну вредност броја. Основна скала очувава вредности у опсегу од -1 до 1. Децимално скалирање може се интерпретирати следећом једначином:

$$v'(i) = \frac{v(i)}{10^k}$$

Где је *v(i)* вредност атрибута *v* у инстанци *i*, а $v'(i)$ његова скалирана вредност за најмању могућу вредност *k* тако да $max(|v'(i)|) < 1$. Поступак је такав да се најпре одреди максимална $|v'(i)|$ вредност за скуп података. Након тога врши се померање децималне тачке све док нова скалирана максимална апсолутна вредност није мања од 1. Затим се делилац примењује на све остале *v(i)* вредности. На пример уколико је највећа вредност у скупу података 589, а најмања вредност -875, тада максимална апсолутна вредност постаје .875, што доводи до тога да се делилац за све *v(i)* вредности атрибута сетује на *1000 (к=3)*.

Min-Max нормализација се користи како би се добила боља расподела вредности над целим нормализованим интервалом. Примена оваквог метода води квалитетнијим резултатима јер, на пример претходни метод може довести до груписања вредности на малом подинтервалу читавог региона. *Min-Max* нормализација на интервалу [0,1] може се интерпретирати следећом једначином:

$$v'(i) = \frac{v(i) - \min(v(i))}{max\big(v(i)\big) - min(v(i))}$$

Слична трансформација се може применити за нормализациони интервал [-1,1]. Минимална и максимална вредност атрибута *v*проналазе се аутоматски на целокупном скупу података или се за исти врши процена од стране експерата у датој области. Сложеност целог поступка није велика ако се узме у обзир да

аутоматско проналажење минималне и максималне вредности захтева још један пролазак кроз цео скуп подака. Одређивање минималне и максималне вредности од стране стручњака може довести до ненамерног гомилања нормализованих вредности.

Нормализација стандардном девијацијом даје добре резултате са мерама растојања, али врши трансформацију скупа података у форму непрепознатљиву са становишта оригиналног скупа података. За атрибут v у скупу података израчунавају се просечна вредности и стандардна девијација. Након тога се за вредност атрибута у инстанци i израчунава нормализована вредност према једначини:

$$v(i) = (v[i] - mean[v])/std(v)$$

На пример, уколико је иницијални скуп вредности атрибута $v = \{1,2,3\}$, tada је средња вредност $mean(v) = 2$, стандардна девијација $std(v) = 1$, док би нови скуп нормализованих вредности изгледао $v^* = \{-1,0,1\}$.

Посматрајући све наведене начине нормализације података, може се закључити да је нормализација као процес значајна за имплементацију неколико различитих data mining метода. Такође, веома је значајно нагласити да нормализација није једнократни или једнофазни процес. Уколико одабрана data mining техника захтева нормализован скуп података, над којим ће се извршавати применом ових начина нормализације, може се добити трансформисани скуп података. Притом у свакој од наредних фаза data miningпроцеса, као и над свим новим подацима који се додају у иницијални скуп податка, идентична нормализација се мора применити. Из тог разлога у процесу имплементације нормализациони параметри се морају чувати заједно са имплементираним решењем.

4.1.2 Редукција димензионалности

Нумерички атрибути у скупу података могу имати најразличитије вредности. Понекад имају онолико различитих вредности колико има инстанци у посматраном скупу података. За већину data mining техника мале разлике између нумеричких вредности нису од превеликог значаја. Међутим, у исто време ове мале разлике у вредностима једног те истог атрибута могу довести до

деградирања перформанси самог метода, као и крајних резултата примене истог. Понекад се оне посматрају као рандом варијације једне те исте вредности. Из тог разлога практично је извршити уједначавање вредности одређене променљиве. Неки од једноставнијих метода сређивања података врше заокруживање података на унапред дефинисани степен прецизности. Уколико се ради о реалним нумеричким вредностима заокруживање истих на одређени број децималних места може представљати један од једноставнијих метода уједначавања података за скупове података са великим бројем узорака, где сваки од узорака поседује сопствену реалну вредност. На пример,уколико је скуп вредности за посматрани атрибут $F = \{0.93, 1.01, 1.001, 3.02, 2.99, 5.03, 5.01, 4.98\}$ тада уједначени скуп вредности изгледа $F_{ujednačeno} = \{1.0, 1.0, 1.0, 3.0, 3.0, 5.0, 5.0, 5.0\}$. Извођење овакве трансформације не доводи до губитка квалитета података у скупу података и у исто време редукује број различитих реалних вредности за посматрани атрибут. Редуковање броја различитих вредности једног те истог атрибута у исто време доводи до редуковања димензионалности простора података. Редуковање димензионалности података резултује бржим израчунавањем при чему се очувава жељена тачност самог процеса. Такође, редуковање броја различитих вредности пружа боље могућности за примену појединих data mining техника.

Поред редуковања димензионалности података редуковањем приближних вредности једног те истог атрибута постоје и други поступци који могу смањити димензионалност скупа података. Уколико се ради о великим скуповима података који у исто време најчешће нису систематски прикупљани како би се над њима примењивале неке од data mining техника, овакви скупови се могу обрадити у циљу елиминације појединих атрибута, што значи да се из скупа података могу избацити небитни, корелациони и редудантни атрибути без деградације перформанси data mining процеса. У суштини, на крају процеса редукције целих атибута потребно је да у скупу података остану само атрибути коју се релевантни за целокупан data mining процес. Очекивано је да data mining апликација обрадом скупа података са овако одабраним атрибутима треба да резултује максималним перформансама и минималном сложеношћу обраде. Управо из ових разлога процес редукције атрибута у скупу података треба да резултује:

- мањим скупом података на основу кога ће обука data mining алгоритма бити једноставнија и бржа;
- већом тачношћу data mining процеса, што утиче на креирање бољег модела на основу података;
- креирањем једноставнијих резултата data mining процеса тако да они буду једноставнији за разумевање и коришћење;
- мањим бројем атрибута у скупу података, што у наредним етапама прикупљања података и допуне скупа података доводи до мањег утрошка времена које би се трошило на елеминацију небитних и погрешних атрибута.

Како су у највећем броју случајева подаци у скупу података представљени табеларно, елиминација непотребних атрибута своди се на елиминацију колона у табели података. Сам проблем уклањања колона нема превелику комплексност, у поређењу са проблемом одабира адекватног критеријума на основу кога ће се одредити које атрибуте треба уклонити из скупа података. У зависности од тога да ли се примењују технике надгледаног или ненадгледаног учења разликују се и технике које се могу применити на редукцију броја атрибута, а самим тим и на редукцију димензионалности података. Технике редуковања димензионалности скупа података могу бити засноване или на трансформисању иницијалног скупа атрибута у нови редуковани скуп атрибута или на селектовању подскупа атрибута из оригиналног скупа. У сваком случају, два стандардна задатка су повезана са креирањем редукованог скупа атрибута и могу се се класификовати као:

Селекција атрибута (енг. *Feature Selection*) - познавањем домена примене финалних циљева data mining процеса, анализом иницијалног скупа података се може извршити селекција подскупа атрибута. Овај процес се може извршити ручно прегледом скупа података и атрибута у скупу података од стране стручњака у датој области. Истовремено, процес селекције атрибута може бити реализован уз помоћ аутоматске процедуре. Уопштено посматрајући, методе селекције атрибута су применљиве у једном од три концептуална *framework*-а: метода филтра, метода омота (енг. *wrapper*) и хибридни метод. Ове три базичне фамилије метода се разликују у начину инкорпорације алгоритма машинског учења у

процес евалуације и селекције атрибута. Применом модела филтра процес селекције атрибута се обавља у предпроцесинг фази data mining процеса, без покушаја директне оптимизације перформанси, што значи да се селекцијом атрибута не врши подешавање скупа података са циљем фаворизовања одређене data mining технике, већ се скуп података редукује и на такав начин се стварају исти услови за будућу примену било које data mining технике [6]. Ово се обично постиже применом евалуационе функције која помоћу метода тражења врши одабир подскупа атрибута који максимизују саму функцију. Процес примене дубинске претраге најчешће је веома тежак за праћење због великог броја атрибута у иницијалном скупу података. Управо из тог разлога све методе филтра користе хеуристику базирану на уопштеним карактеристикама скупа података, а не на конкретном алгоритму машинског учења. Применом метода филтрирања атрибута, током времена издвојили су се различити алгоритми који се могу применити у различитим ситуацијама у зависности од каракеристика скупа подата. Један од алгоритама који се може применити у циљу проналажења минималне комбинације атрибута на основу којих се иницијални скуп података може поделити на чисте класе јесте *FOCUS* [42]. Примена овог алгоритма као резултат даје подскупове атрибута креиране тако да свака комбинација вредности атрибута одговара и указује на једну јединствену класу података. На крају процеса селекције атрибута, применом овог алгортима, крајњи подскуп атрибута може се проследити ID3 алгоритму машинског учења, што ће довести до ефикасног креирања стабла одлучивања. Поред овог алгоритма филтрирања атрибута могу се издвојити и други алгоритми. Неколико истраживача су радила на истраживању могућности примене више алгоритама машинског учења у циљу филтрирања атрибута. Тачније, истраживачи су користили конкретан алгоритам машинског учења као предпроцесор за издвајање подскупа корисних атрибута из иницијалног скупа података. Овако креирани подскуп података користе други алгоритми машинског учења у фази обраде податка. На пример, стабла одлучивања се могу користити као алгоритми филтрирања атрибута и креирања подскупа атрибута од интереса за даљу обраду од стране алгоритама машинског учења. С тим у вези, у једном од истраживања *C4.5* алгоритам машинског учења је коришћен за креирање стабла одлучивања на основу иницијалног скупа

података [43]. Атрибути који су се након обраде овим алгоритмом појавили у коначном стаблу одлучивања коришћени су касније у процесу обраде скупа података помоћу алгоритма *к*-најближих суседа. Коришћење оваквог система резултовало је далеко бољим перформансама од коришћења, како C4.5 тако и алгоритма *к*-најближих суседа понаособ. На сличан начин су аутори у једном од истраживања користили стабла одлучивања како би извршили филтрирање и селекцију атрибута од интереса на основу којих је касније вршено креирање *Bayesian* неуронске мреже [44].

За разлику од метода филтра, *wrapper* методи редукују скуп атрибута како би иницијални скуп атрибута прилагодили будућој примени одређене технике машинског учења. *Wrapper* методи врше дељење иницијалног скупа атрибута у подскупове атрибута. Након дељења врши се процена сваког од подскупова атрибута понаособ на основу очекиваних перформанси data mining технике машинског учења. Најчешће се евалуација сваког појединачног подскупа атрибута врши помоћу индукционих data mining алгоритама. *Wrapper* методи селекције атрибута дају боље резултате у односу на методе филтра због саме чињенице да селекцију атрибута прилагођавају према специфичној интеракцији између индукционог алгоритма и података у иницијалном скупу података. Са друге стране, ови методи имају знатно веће време извршења у односу на методе филтра због саме чињенице да се морају позивати изнова и изнова са сваким коришћењем другог data mining алгоритма [45]. Како је *wrapper* метод добро дефинисан процес, већина варијација у његовој примени узрокована је одабиром метода којим се врши процена тачности одабраног индукционог data mining алгоритма, као и начином организације претраге. Како су *wrapper* методе засноване тако да се прилагођавају конкретном data mining алгоритму који ће се користити у даљем процесу обраде података, могу се према начину рада data mining метода издвојити и *wrapper* методе селекције атрибута. Прва од група *wrapper* метода обухвата методе оријентисане ка data mining техникама базираним на стаблима одлучивања. Истраживања базирана на примени ове методе имају дугу историју примене [46]. *ID3* и *C4.5* алгоритми коришћени су као *wrapper*-и за селекцију атрибута у три вештачка и три природна скупа података. Процена тачности је вршена применом 25-*fold* унакрсне валидације над тренинг скупом

података. Резултати истраживања су показали да селекција атрибута није значајно утицала на опште перформансе *ID3* и *C4.5* алгоритама [47]. Основни ефекат селекције атрибута применом ових метода била је редукција величине стабала креираних помоћу ова два алгоритма. Друга од група *wrapper* метода обухвата методе оријентисане ка data mining техникама базираним на раду са инстанцама. У ову групу се сврставају класификационе data mining технике. Један од алгоритама под називом *OBLIVION* користи елиминацију атрибута у стаблу одлучивања почевши од последњег чвора ка корену. Током креирања стабла одлучивања од иницијалног сета атрибута сваком од атрибута у поступку класификације додељен је одређени број претпоставки [45]. Оваква структура је еквивалентна методу најближег суседа. Експерименти у којима је коришћен *OBLIVION* алгоритам коришћењем *k-fold* унакрсне валидације над неколико вештачких домена показали су да је могуће уклонити редундантне атрибуте, као и да је овакве атрибуте могуће брже обучити него *C4.5*. Ово је посебно уочљиво у домену скупова података у којима атрибути међусобно интерагују. Такође, *wrapper* селекција атрибута се може искористити како би се одредио потенцијал *DTM* (енг. *Decision Table Majority*) класификатора. Одређене структуре података дозвољавају коришћење брзих инкременталних унакрсних валидација заједно са *DTM* класификаторима. Експерименти показују да се *DTM* класификатори коришћењем одговарајућег подскупа атрибута могу ставити у паралелу са веома софистицираним алгоритмима као што је примера ради *C4.5* алгоритам [48]. Трећа од група *wrapper* метода обухвата методе оријентисане ка data mining техникама базираним на *naive Bayes* класификационим алгоритмима. Претпоставка на којој се базира употреба *naive Bayes* класификатора је да су за сваку од класа вероватноће дистрибуције атрибута независне једна од друге. Уклањањем редундантних атрибута из скупа података могу се повећати перформансе класификатора. Код *naive Bayes* класификатора, за разлику од стабала одлучивања и других data mining класификационих алгоритама, спроводи се претрага од почетка ка крају скупа атрибута. На овакав начин се врши издвајање скупа атрибута над чијим ће се вредностима базирати креирање модела и рад *naive Bayes* класификатора [45]. Употреба овакве претраге омогућава моментално откривање редундантних атрибута током процеса додавања нових

атрибута у скупу података. У неким од експеримената примењивана је *hill climbing* стратегија претраге на отривање редундатних атрибута. Као у претходном случају и ова стратегија даје позитивне ефекте на перформансе *naive Bayes* класификатора, како приликом додавања појединачног новог атрибута тако и приликом додавања атрибута који је остао неселектован у претходним фазама селекције [49].

Хибридне методе врше комбиновање различитих метода селекције атрибута и алгоритма машинског учења у јединствену формулацију проблема оптимизације иницијалног скупа података. У зависности од броја инстанци у скупу података, као и у зависности од димензионалности скупа података применом хибридног метода, врши се одабир између метода филтра и *wrapper* метода. Велики број инстанци података и велика димензионалност, као и ефикасност израчунавања и неутралност према data mining алгоритмима машинског учења, најчешће су главни критеријуми за одабир метода филтра.

Издвајање или трансформација атрибута (енг. *Feature Extraction/Transformation*) - Током процеса обраде иницијалног скупа података, могу се применити трансформације које имају изненађујуће велики утицај на резултате примене data mining метода. У овом смислу, композиција и/или трансформација атрибута у скупу података је одлучујући фактор у квалитету резултата data mining процеса. У већини инстанци података, композиција атрибута је зависна од познавања примене скупа података. Такође, интердисциплинарни приступ проблему композиције атрибута у инстанцама података обезбеђује значајна побољшања у процесу припреме података. Ипак, неке од техника опште намене као што је *PCA* (енг. *Principal Component Analysis*) се често користе са великим успехом. *PCA* је практично математичка процедура која се може користити за редукцију великог скупа података у мањи скуп. Редукција димензионалности применом ове процедуре остварује се трансформацијом одређеног броја међусобно повезаних атрибута у мањи број међусобно неповезаних арибута. Практично улагањем минималног напора *PCA* обезбеђује шему која показује како редуковати комплексан скуп података на мању

димензију откривањем једноставније структуре која је често подцењена или неуочљива [50].

Математичка позадина израчунавања *PCA* коефицијената за разлику од већ поменутих техника за редукцију димензионалности захтева да скуп података мора да буде уређен у форми матрице оријентисане у форми колона, тако да су инстанце података представљене по колонама, док сваки од редова одговара атрибуту података. На почетку се врши израчунавање средње вредности сваког од нумеричких атрибута. Уколико је скуп података означен са *D*, процес израчунавања може бити представљен следећом једначином:

$$\overline{D}_1 = \frac{1}{n}\sum_{i=1}^{n} D_{1i}$$

$$\overline{D}_2 = \frac{1}{n}\sum_{i=1}^{n} D_{2i}$$

$$\vdots$$

$$\overline{D}_m = \frac{1}{n}\sum_{i=1}^{n} D_{mi}$$

Где је *m* број редова у матрици *D* (број атрибута), а *n* број колона у матрици *D* (број инстанци). Добијени резултати се морају одузети од сваке од иницијалних вредности у матрици, што резултује креирањем нове централизоване матрице података D'. Поменутим одузимањем ће се извршити трансформација података у нови координатни систем транслирањем њиховог координатног почетка у центар или у средњу вредност података. Следећи корак у PCA трансформацији је израчунавање коваријансе. С обзиром да су подаци организовани у матричну форму, коваријанса се може израчунати применом метода множења матрица, као што је приказано следећом једначином:

$$C = \frac{1}{n} D' {D'}^T$$

Коваријанса променљивих са самим собом једнака је варијанси дате променљиве. Елементи матрице коваријансе који одговарају главној дијагонали представљају варијансу појединачних променљивих. Овако креирна матрица коваријансе значајна је за будућу анализу. На основу израчунате матрице коваријансе даље се израчунавају њене сопствене вредности и сопствени вектори. Сопствене вредности матрице коваријансе представљају корен њеног карактеристичног полинома. Сопствени вектор линеарне трансформације је ненулти вектор који не мења свој смер након примене линеарне трансформације [51]. Сопствне вредности се могу израчунати као:

$$\det(\lambda I_n - C) = 0,$$

Где I_n представља јединичну матрицу, а λ представља сопствене вредности које требају да се израчунају. Након израчунавања сопствених вредности, врши се израчунавање сопствених вектора тако што се проналази ненулто решење једначнице сопствених вредности.

$$(\lambda I_n - C)\vec{v} = \overrightarrow{0},$$

где $\vec{v}$ представља сопствене векторе који требају да се израчунају.

У пракси се оваква реализација претвара у систем линеарних једначина са познатим коефицијентима. Израчунати сопствени вектори се морају поставити један до другог како би формирали колоне квадратне матрице W. Формирана матрица W је ортогонална матрица која представља трансформациону матрицу координата. Како би се добила крајња форма матрице W врши се њено транспоновање, што доводи до креирања матрице W^T. Нове координате за иницијални сет података се добијају извођењем коначне операције множења, као што је приказано следећом једначином:

$$D_{PCA} = w^T D'$$

Овако добијене нове координате називају се главне компоненте (енг. *Principal Components).* У општем смислу *PCA* користи основне варијанте кодирања у структури података, а затим сопствене векторе како би се одредио нови скуп координата које најбоље откривају структуру података проналажењем

одговарајућег скупа праваца. Значајност креиране D_{PCA} матрице се огледа у томе да се помоћу ове матрице скуп података велике димензионалности редукује на само неколико главних компоненти за које се очекује да задрже већину варијација у подацима, омогуће правилну визуелизацију и издвајање значајних патерна и структура.

Уколико се упореде метод селекције и метод издвајања и трансформације са аспекта задржавања оригиналног значење атрибута, као и уколико је потребно одредити који од атрибута су најбитнији, метод селекције ће дати боље резултате у односу на метод издвајања и трансформације [6]. У погледу сложености, метод селекције се одликује мањом сложеношћу, с обзиром да се након одабира атрибута од интереса врши само прикупљање и обрада истих, док се код метода издвајања и трансформације мора извршити редукција димензионалности за све улазне атрибуте.

4.1.3 Разлика и стављање у однос (енг. *Differencies and Ratios*)

Чак и мале промене у атрибутима података могу довести до значајних побољшања у data mining перформансама. Ефекти релативно малих трансформција улазних или излазних атрибута нарочито су значајни приликом спецификације data mining циљева. Две врсте једноставних трансофрмација под називом разлика и однос могу бити од користи приликом спрецификације циљева, посебно уколико се примене на излазне атрибуте. Поменуте трансформације у многим случајевима креирају боље резултате него што је то случај са рецимо простим почетним циљем предикције нумеричке вредности. За већину data mining техника мањи број алтернатива доводи до унапређења ефикасности самог алгоритма и у највећем броју случајева ће дати боље резултате. Разлика као трансформација излазних атрбута може умањити број различитих вредности излазних атрибута применом просте трансформације. На пример, уместо да се као излазни атрибут користи вредности *s(t+1),* много ефикасније може бити коришћење излазне вредности атрибута у облику *s(t+1)-s(t).* Са друге стране, однос као још једна од трансформација излазних атрибута значи да ниво повећања или смањења вредности излазних атрибута може унапредити перформансе целог data mining процеса. С тим у вези, вредности излазних атрибута се могу ставити у

однос, те ће се самим тим и њихова различитост повећати или смањити. Тако уместо коришћења вредности атрибута у облику *s(t+1)* може се користити вредност у облику *s(t+1)/s(t).* Трансформације попут разлике и односа нису само корисне по питању уређивања нумеричких вредности излазних атрибута, већ и за улазне атрибуте [6]. Могу се користити за уређивање промена у времену једног атрибута или композиције различитих улазних атрибута. У многим медицинским скуповима података користе се тежина и висина пацијента као два основна улазна атрибута за дијагностификовање болести и праћење стања пацијента. Многе апликације показују боље перформансе приликом дијагностификовања, уколико користе само један атрибут под називом маса тела који представља однос између тежине и висине, уместо ова два атрибута засебно.

Логичке трансформације се могу искористити и у циљу креирања нових атрибута. Понекад је корисно креирати нови атрибут који ће одредити логичку вредност релације A>B креиране између постојећих атрибута A и B. Битно је истаћи и да не постоје универзално најбоље методе трансформација података и да много тога зависи од самих података. Као што је већ напоменуто посебна пажња треба да буде усмерена на композитне атрибуте, јер некада проста трансформација може бити далеко ефикаснија у погледу крајњих перформанси него што би то био случај са преласком на неку другу data mining технику.

4.1.4 Дискретизационе методе

Дискретизација је процедура обраде података која врши трансформацију квантитативних података у квалитативне податке. Data mining технике често раде са квантитативним податацима, међутим чак и када су data mining алгоритми оријентисани на рад са великим квантитативним подацима, процес учења је далеко ефикаснији и ефективнији уколико се ради са квалитативним подацима. Из тог разлога, дискретизација је веома популарна, као једна од трансформационих техника. Током времена примене предложено је много дискретизационих техника. Евалуција ових алгоритама показала је да дискретизација помаже код повећања перформанси алгоритма учења као у код разумевања резултата процеса учења [52]. Дискретизација врши трансформацију једног типа података у други тип података. Уколико се посматра таксономија дискретизационих метода, може се

уочити да постоје различити аспекти различитости између дискретизационих метода. Уобичајено је да дискретизационе методе могу бити или примарне или композитне. Примарне методе извршавају процес дискретизације без референцирања на неку другу дискретизациону методу, док су композитне методе изграђене на основу неке друге методе или више других. Композитне методе најпре врше одабир примарне дискретизационе методе како би одредиле иницијалну тачку пресека. Након тога се фокусирају на прилагођавање иницијалног скупа пресечних тачка остварењу основног циља. Таксономија композитних метода је веома флексибилна и у исто време зависна од таксономије примарне дискретизационе методе. Са друге стране, за разлику од композиних метода примарне методе имају јасно дефинисану таксономију, те се према њој могу класификовати као:

- Надгледано наспрам Ненадгледаног - посматрајући из контекста дискретизационих метода, методе које користе информације садржане у класним атрибутима тренинг инстанци за селекцију дискретизационих тачака пресека називају се надгледане. Методе које не користе класне информације су ненадгледане. Надгледане методе даље се могу поделити на методе базиране на грешкама, методе базиране на ентропији и методе базиране на статистици, све у зависности од тога да ли интервали селектовани коришћењем метрике базиране на грешкама у тренинг подацима, ентропији интервала података или на некој статистичкој мери.
- Параметриске наспрам Непараметриских (енг. *Parametric vs. Non-parametric*) - параметријска дискретизација захтева од корисника да унесе максималан број дискретизационих интервала. Непараметриске дискретизационе методе користе само информације из скупа података и не захтевају никакав унос од стране корисника.
- Хијерархијске наспрам нехијерархијских (енг. *Hierarchical vs. Non-hierarchical*). Хијерархијске дискретизационе методе врше селекцију тачака пресека током инкременталног процеса креирајући на тај начин имплицитну хијерархију над опсегом вредности. Процедуре које се примењују у процесу хијерархијске дискретизације су дељење или спајање [53]. Неке од дискретизационих метода могу користити обе процедуре.

Нехијерархијске методе дискретизације не врше формирање било какве хијерархије током процеса дискретизације. На пример многи методи врше скенирање вредности атрибута само једном и на такав начин врше секвенцијално формирање интервала.

- Једноваријантне наспрам Вишеваријантних (енг. *Univariate vs. Multivariate*) - методе које врше дискретизацију сваког атрибута засебно, без узимања у обзир његових потенцијаних веза са другим арибутима су једноваријантне. Мотоди који узимају у обзир везе имеђу атрибута током процеса дискретизације називају се вишеваријантни методи [54].
- Раздвајања наспрам Нераздвајања (енг. *Disjoint vs. Non-disjoint*) - методе раздвајања врше дискретизацију опсега вредности атрибута током процеса дискретизације у раздвојеним интервалима, при чему не долази до преклапања интервала. Методе нераздвајања дискретизују опсед вредности у интервале који се могу преклапати [55].
- Глобалне наспрам Локалних (енг. *Global vs. Local*) - глобалне методе врше дискретизацију узимајући у обзир целокупни простор скупа података. Дискретизација применом глобалних метода се врши само једном и то коришћењем једног скупа интервала током једноставних задатака класификације. Локалне методе дискретизације дозвољавају формирање различитих скупова интервала за један атрибут, при чему се сваки скуп примењује у другачијем класификационом контексту. На пример, различита дискретизација једног те истог атрибута се може применити у различитим чворовима стабла одлучивања.
- Брзе наспрам Лењих (енг. *Eager vs. Lazy*) - брзе методе изводе процес дискретизације пре саме класификације. Лење методе изводе процес дискретизације током процеса класификације [56].
- Временски осетљиве наспрам Временски неосетљивих (енг. *Time-sensitive vs. Time-insensitive*) - током извођења временски осетљиве дискретизације квалитативне вредности повезане са квантитативним вредностима се могу заменити, штозначи да се једна квантитативна вредност може дискретизовати у различите вредности, у зависности од претходних вредности у временској серији. Временски неосетљиве методе

дискретизације користе само стационарна својства квантитативних података.

- Редне наспрам Номиналних (енг. *Ordinal vs. Nominal*) - редне дискретизационе методе врше трансформацију квантитативних података у редне квалитативне подаке. Циљ је искористити предности информација имплицитно у квантитативним атрибутима тако да се рецимо вредности 1 и 2 не разликују као вредности 1 и 10. Номиналне дискретизационе методе врше трансформацију квантитативних података у номиналне квалитативне податке. Из тог разлога је информација о редоследу одбачена.
- Fuzzy vs. Non-fuzzy - *fazzy* diskretizacione методе најпре дискретизују квантитативне вредности атрибута у интервале. Након тога постављају неку врсту функције припадноси на свакој тачки пресека као фазну границу. Функција припадности врши мерење степена припадности сваке вредности одређеном интервалу. Постављањем *fazzy* граница конкретна вредност у исто време може бити дискретизована у неколико различитих интервала са различитим степенима припадности [57]. *Non-fazzy* diskretizacione metode formiraju striktne granice bez korišćenja funkcije pripadnosti.

На основу наведне таксономије могу се издвојити неке од основних дискретизационих метода које се могу наћи у примени. Иако су различите дискретизационе методе присутне у пракси од посебног значаја за ово истраживање су дискретизационе методе оријентисане ка машинском учењу. Различити типови машинског учења имају различите карактеристике, те самим тим захтевају различите стратегије дискретизације. Одабир дискретизационе методе зависи од контекста алгоритма машинског учења који ће се примењивати над подацима, јер не постоји универзална дискретизациона метода која се може применити независно од тога која ће се техника учења користити. Тако алгоритми машинског учења базирани на стаблима одлучивања могу имати лошије перформансе због проблема фрагментације, док са друге стране могу остварити већи успех у односу на остале алгоритме машинског учења уколико се спроведе дискретизација која за резултат има неколико интервала. Једна од популарних дискретизационих метода намењена стаблима одлучивања је вишеинтервалска-

минимизација-ентропије (енг. *Multi Interval Entropy Minimization - MIEMD*). Ова метода врши дискретизацију квантитативних атрибута израчунавањем ентропије информације у класном атрибуту, као да ће класификација користи само тај један атрибут након процеса дискретизације. Овакав метод може бити користан за „подели и владај“ стратегију стабла одлучивања. У исто време, не мора да буде одговарајући за остале механизме машинског учења као што је рецимо *naive Bayes* класификатор [55]. За стабла одлучивања веома је важно минимизовати број различитих вредности једног атрибута, како би се избегао проблем фрагментације. Уколико атрибут има много вредности, даље креирање стабла од тог атрибута резултоваће великим бројем потега, од којих ће сваки представљати свега неколико инстанци. Овакав начин креирања стабла довешће до проблема приликом селекције одговарајућег теста.

Уколико се посматрају *naive Bayes* класификатори, одговарајуће дискретизационе методе могу бити дискретизација фиксне фреквенције и метода дискретизације нераздвајања. *Naive Bayes* класификациони алгоритам машинског учења претпоставља да су атрибути независни један од другог за дату класу, па самим тим и није кандидат за проблем фрагментације, као што је то случај са стаблима одлучивања. *Naive Bayes* алгоритам машинског учења је посебно популаран за велике скупове података управо због своје ефикасности.

4.2 Обрада недостајућих вредности података

Улазни подаци, припремљени за обраду data mining техникама, обично су организовани у форми скупа података или, прецизније, табеле одлучивања, што значи да су инстанце података сачињене од независних променљивих (атрибута) и зависних променљивих (одлука или класних атрибута). Вредности појединих аутрибута у скупу података могу недостајати из различитих разлога, од оних мање битних да конкретни атрибут није био од интереса током прикупљања података, до чињенице да је дошло до његовог изостављања приликом уписа или његовог случајног брисања. Уколико се ради о падатку који је постојао, његово постојање је битно, како приликом обраде помоћу статистичких техника тако и приликом обраде применом data mining техника. Из тог разлога, припрема скупа података у фази предпроцесирања укључује коришћење комплексних

статистичких метода или коришћење специфичних data mining алгоритама за решавање проблема недостајућих вредности. Генерално гледано, методи решавања недостајућих вредности атрибута могу се сврстати у две групе: секвенцијални и паралелни методи. Секвенцијалне методе пропагирају решавање недостајућих вредности у предпроцесинг фази, док се применом паралелних метода недостајуће вредности атрибута одређују током главног процеса обраде података и откривања корисног знања. Секвенцијалне методе обухватају следеће технике обраде инстанци података са недостајућим вредностима атрибута:

- **Брисање инстанци података које садрже недостајуће вредности** - применом овог метода бришу се све инстанце у скупу податка које садрже атрибут без вредности. Овакав метод може довести до значајног смањења скупа података над којим ће се у каснијим фазама примењивати data mining методе учења. Међутим, постоје одређени разлози зашто се овакви методи сматрају добрим [58], [59].
- **Најчешћа вредност атрибута** - ово је један од можда најпростијих метода замене недостајуће вредности. Атрибуту који нема вредност се додељује вредност која се најчешће појављује за исти атрибут у другим инстанцама истог скупа података. Пример примене оваквог метода је имплеменација у [60].
- **Најчешћа вредност атрибута органичена на део скупа података** - ово је модификација претходне методе, којом се за недостајућу вредност атрибута, такође, узима вредност која се најчешће појављује, с тим што се најфрекветнија вредност не тражи над целим скупом већ над његовим унапред одабраним делом. Разлика је у томе што се у одабраном делу за претрагу могу наћи и недостајуће вредности. Издвајање подскупа у коме ће се вредност тражити не мора бити секвенцијално већ нови скуп података могу креирати рандом инстанце података оригиналног скупа.
- **Додељивање свих могућих вредности недостајућој вредности атрибута** - овим методом се свака инстанца података са недостајућом вредношћу атрибута мења сетом нових инстанци у којима је вредност недостајућег атрибута замењена сваком од вредности истог атрибута које се појављују у скупу података. Оваквим методом долази до увећања скупа података, што

може довести до контрадикторности. Контадикторност се огледа у томе да се може јавити већи број инстанци података са истим вредностима за независне променљиве, а различитим вредностима зависне променљиве (класног атрибута). У сваком случају, применом различитих техника могу се индуковати предикциона правила из контадикторног скупа податка [61].

- **Замена вредности недостајућег атрибута просечном вредношћу** - овакав метод се користи код решевања проблема недостајућих нумеричких атрибута. Замена недостајуће вредности се може извршити просечном вредношћу датог атрибута у осталим инстанцама података. Овај приступ је применљив када се ради о проблему класификације где су узорци унапред класификовани. Уколико у скупу података постоје симболички атрибути (текстуални, описни, номинални) са недостајућим вредностима примена ове методе се своди на замену недостајуће вредности најфреквентнијом вредношћу атрибута.
- **Замена вредности недостајућег атрибута просечном вредношћу одређеном над делом скупа** - примена овог метода је потпуно иста као у претходном случају, с тим што се проналажење просечне вредности атрибута ограничава на део скупа података уместо на цео скуп. Нови скуп података на основу кога се одређује просечна вредност може бити креиран узимањем одређеног броја инстанци података из оригиналног скупа секвенцијално или рандом. Такође, примена овог метода на симболичке вредности атрибута се своди на проналажење најфреквентније вредности у ново креираном скупу података који представља део оригиналног скупа.
- **Метод глобалног најближег поклапања** - Овај метод се базира на замени вредности недостајућег атрибута вредношћу истог атрибута у другом скупу података који је најсличнији могући оригиналном скупу података. У тражењу најближег поклапања врши се поређење два вектора са вредностима атрибута. Први вектор је вектор са атрибутима којима недостају вредности, док други вектор представља вектор са вредностима који су кандидати за могуће поклапање. Тражење замене се спроводи за све случајеве. За сваки од случајева израчунава се растојање. Случај за који је растојање најмање узима се као вредност која ће заменити недостајућу

вредност атрибута. Уколико су x и y елементи вектора, дистанца између њих може се израчунати према следећој једначини:

$$distance(x, y) = \sum_{i=0}^{n} distance(x_i, y_i), \text{ где је}$$

$$distance(x_i, y_i) = \begin{cases} 0 & za\ x_i = y_i \\ 1 & ako\ su\ x\ i\ y\ simboli\ i\ x_i \neq y_i, ili\ je\ x_i = ?\ ili\ y_i = ?, \\ \dfrac{|x_i, y_i|}{r} & ako\ su\ x_i\ i\ y_i\ brojevi\ i\ x_i \neq y_i \end{cases}$$

где је r разлика између максималне и минималне вредности познатих нумеричких атрибута са недостајућим вредностима. Уколико се приликом израчунавања јаве две исте вредности за дистанце мора се применити нека хеуристика. На пример, може се одабрати први случај.

- **Метод глобалног најближег поклапања ограничен на део скупа** - Као и ранијим методама ограниченим на део скупа и овај метод је сличан методу глобалног поклапања с разликом ограничења на део скупа. Најпре се оригинални део скупа података заједно са атрибутима којима недостају вредности дели на мале подскупове података. Сваки од малих подскупова податка одговара делу скупа који се посматра као целина. Након поделе скупа података, исти метод глобалног најближег поклапања се појединачно примењује на оба скупа података. Оба процесирана скупа података или табеле података се евентуално спајају у један.

Паралелне методе проблем недостајућих вредности атрибута решавају креирањем индукционих правила. Овде се улазни подаци не предпроцесирају на исти начин као код секвенцијалних метода, већ се алгоритам који ће се користи за одређивање недостајућих вредности обучава користећи оригинални скуп података заједно са инстанцама које садрже недостајуће вредности. Овде се могу применити неки од одговарајућих data mining алгоритама надгледаног учења. Практично, недостајуће вредности се могу добити предикцијом заснованом на новом скупу података у коме је атрибут са недостајућим вредностима овог пута класни атрибут. За овакав скуп података креирају се правила која ће описивати везу између података и класног атрибута.

4.3 Детекција и отклањање *outlier*-а

Анализом података и отклањањем недостајућих вредности врши се отклањање великог броја могућих грешака. Међутим, ове методе не могу указати на много сложеније грешке које се јављају у већини великих скупова података. Грешке које укључују међусобне односе између једног или више поља у скупу података су најчешће најтеже за детекцију и уклањање. Овакви типови грешака захтевају знатно подробнију инспекцију и анализу података. Проблем детекције и уклањања оваквих грешака назива се детекција outlier-а. На пример, уколико се преко 99.9% података уклапа у генералну форму или нумерички опсег, онда се може смарати да преостали део података 0.1% представља кандидате за могућу грешку. Информација о томе како подаци требају да изгледају или у каквом опсегу се требају наћи може допринети откривању грешака. У пракси подаци из стварног света су најчешће разнолики и најчешће се не уклапају у ниједну стандардну статистичку расподелу. Проблем је посебно изражајан када се подаци посматрају кроз неколико димензија. Такође, са повећањем комплексности података, типа додавањем све више и више атрибута, увођењем номиналних вредности у скуп података, као и дефинисањем класних атрибута путем номиналних вредности детекција *outlier*-а постаје све сложенија [62]. У таквим случајевима за детекцију и отклањање свих *outlier*-а потребно је применити неколико метода. Посматрано из угла метода које се могу применити на проблем детекције и отклањања *outlier*-а, могу се извојити статистичке методе, различите data mining методе, методе засноване на креираним шаблонима између података, као и асоциациона правила.

Вредности *outlier*-а за одређена поља у скупу података могу се идентификовати на основу израчунате статистике. Практично, израчунавањем средње вредности, стандардне девијације, опсега вредности, у склопу примене Чебишевљеве теореме они записи који имају вредности изван стандардне девијације у односу на средњу вредност су кандидати за *outlier*-е. Степен одступања од стандардне девијације је прилагодљив [63]. За свако од поља интервали поверења се морају размотрити. Вредност атрибута f_i у инстанци r_i може се сматрати outlier-ом уколико је $f_i > \mu_i + \varepsilon\sigma_i$ или уколико је $f_i < \mu_i - \varepsilon\sigma_i$, где је μ_i средња вредност за атрибут f_i, σ_i стандардна девијација, а ε корекциони фактор дефинисан од стране корисника.

Вредност ε с обзиром да се дефинише од стране корисника мора се базирати на неком домену или познавању вредности података у оквиру скупа података [64]. Истраживања су показала да се опсег вредности које ε може узети креће од 3 до 6. Овакав приступ отклањања outlier-а је најједноставнији али може генерисати лажно позитивне резултате. Из тог разлога требало би се користити у комбинацији са другим методама.

Поред овог метода, под групом статистичких метода могу се издвојити још два метода. Први од њих је откривање *outlier*-а на основу расподеле података. У овом методу подаци добијени нормалном или Поисоновом расподелом података се бирају као најбољи подаци од којих је расподела зависна. *Outlier* се детектује на основу расподеле вероватноће. Метод детекције *outlier*-а базиран на општим шаблонима између тачака података може се сматрати веома ефикасним [65]. Други метод отклањања *outlier*-а базиран је на дубини. Практично овим методом подаци се посматрају као тачке у простору и додељује им се дубина А. *K*-димензионални простор се користи за представљање сваког објекта података у складу са додељеном дубином. За објекте података који имају малу дубину постоји већа вероватноћа да они представљају *outlier*-е. Овакав приступ заснован на дубини сматра се бољим од многих других приступа јер решава проблем расподеле података. Овакав метод омогућава обраду мултидимензионалних објеката података [66]. Основни принцип се базира на израчунавању *K*-димензионалног конвексног простора. Иако се концептуално овај метод може користити за податке са много димензија, практично није применљив на исте. Анализа резултата добијених оваквом методом може се извршити применом визуелизационих техника.

Идентификовање инстанци које садрже *outlier*-е може се извршити и применом кластеризационих техника, примера ради одређивањем Еуклидовог или неког другог растојања. Неки од кластеризационих алгоритама могу пружити веома добру подршку другим методима за детекцију и отклањање *outlier*-а. Ефикасност кластеризационих техника у погледу детекције *outlier*-а огледа се у у томе што су подаци из скупа података применом ових техника подељени у више кластера. Поступак кластеризовања се може поновити више пута све док се не добију

одговарајући кластери у којима је најпогодније детектовати *outlier*-е. Истраживања показују да након неколико извршења над истим скупом података већа гранична вредност за максимално растојање између кластера даје простор за бољу детекцију *outlier*-а. Бржи кластеризациони алгоритми који омогућавају аутоматско подешавање максималне величине кластера, као и скалабилност већих скупова података погоднији су за коришћење [64]. Редуковање величине података, применом кластеризационих техника, може се остварити уколико се одабере подскуп на основу кога ће се извршити одабрани алгоритам кластеризације. Примена кластеризационих теханика, са циљем детекције *outlier*-а у тест скупу података, има посебне карактеристике с обзиром да тест скуп података садржи, како доста празних вредности (зависне променљиве које треба одредити) тако номиналне вредности. Утврђивање тога која од номиналних вредности представља *outlier* разликује се од истог поступка примењеног над нумеричким вредностима. Над тест скупом података може се применити кластеризациони алгоритам са дефинисаним Хамингтоновим (енг. *Hamming*) растојањем [67]. Низ нула и јединица може се додати свакој од инстанци. Сваки низ има онолико елемената колико има атрибута у датој инстанци података. Хамингтоново растојање се користи како би се инстанце кластеризовале у групе сличних. Коришћење оваквог растојања неће довести до детекције *outlier*-а, али ће креирати кластере инстанци који се применом других метода могу брже обрадити и у њима детектовати *outlier*-е и евентуално недостајуће вредности. Као што је већ наведено, различити алгоритми кластеризације могу се користити за потребе детектовања *outlier*-а о чему сведоче и спроведена истраживања. *K-means* и *k-median* методе коришћене су за отклањање *outlier*-а у *striming* подацима [63], док су методе као што су *PAM* (енг. *Partitioning Around Medoids*), *CLARA* (енг. *Clustering LARge Applications*) и *CLARANS* (енг. *Clustering Large Aplications на основу Randomized Search*) коришћење за детекцију *outlier*-а као методе раздвајања базиране на центроидалном кластеризовању [68]. Хијерархијско кластеризовање коришћено је за потребе детектовања *outlier*-а при чему је скуп података дељен на мале подскупове [69].

Методи просторне анализе су често блиско повезани са методима за кластеризацију. Просторне аномалије се могу дефинисати као просторно-

референцирани објекти чије се вредности не-просторних атрибута знатно разликују од вредности које за исте атрибуте имају његови суседи. Методи просторне статистике се могу класификовати у две подкатегорије: квантитативни тестови и графички приступ. Квантитативне методе се користе у креирању тестова помоћу којих се врши раздвајање просторих *outlier*-а од остатка података. Графичке методе се заснивају на визуелизацији просторних података којом се означавају *outlier*-и. Примена просторне анализе, у циљу детекције *outlier*-а, значајна је у оним областима у којима просторне информације имају велику улогу, као што су екологија, географски информациони системи, транспорт, метеорологија, климатологија итд [70].

Методе откривања *outlier*-а засноване на идентификованим шаблонима између инстанци података и атрибута у оквиру скупа података обједињују различите технике (дељење, класификација и кластеризација) у циљу идентификовања шаблона који ће карактерисати већину инстанци података. Шаблон се креира тако што групише инстанце које имају сличне карактеристике или слично понашање за процентуално одређени број атрибута у скупу података. Процентуалну вредност може дефинисати сами корисник, и ова вредност се обично поставља на преко 90%. Уколико се користи кластеризовање, као један од метода утврђивања сличности, шансе за идентификовање шаблона се повећавају, с обзиром да инстанце у кластерима имају одређени степен сличности и приближно исти број празних атрибута [71]. На основу шаблона свака од вредности атрибута или инстанца података која се не уклапа у дати шаблон може се сматрати *outlier*-ом. Како су подаци из реалног живота најчешће неуређени, процес проналажења шаблона између таквих података у скупу података постаје сложенији. У оваквим случајевима се дефинишу посебне вредности мере које се користе за детектовање *outlier*-а као скупа података који показује мању припадност најфреквентнијем уоченом шаблону на основу дефинисане мере [72].

Асоцијациона правила пружају високу поузданост и добру подршку у дефинисању различитих типова шаблона међу подацима. Као и код примене самих шаблона, инстанце података или појединачни атрибути које не прате примењена правила сматрају се *outlier*-има.

Снага асоцијационих правила се огледа у томе да се она успешно могу применити над различитим типовима података. У зависности од тога која асоцијациона правила се примењују зависиће и количина квантитативних и квалитативних информација које ова правила пружају [73]. Термин асоцијациона правила први пут је примењен у контексту анализе потрошачке корпе [74]. У литератури оваква асоциациона правила називају се класична или *Boolean* асоцијациона правила. Сами концепт је проширен у другим студијама и експриментима. Такође, за примену откривања *outlier*-а у скупу података успешно се могу користити квантитативна асоциациона правила (енг. *Quantitative Association Rules*), као и правила односа (енг. *Ratio-Rules*). Ова правила се додатно могу модификовати што као резултат даје ефикаснију примену у детектовању *outlier*-а. Процес детекције *outlier*-а у скупу података применом ординалних правила асоцијативности састоји се из два корака.

- Први корак је пронаћи ординална правила асоцијативности са унапред дефинисаним минималним поверењем. Ово се може урадити на различите начине. На пример, један од начина је примена варијације *apriori* алгоритма.
- Други корак је детекција података у скупу података који крше креирана асоцијациона правила и који се могу узети у обзир као outlier-и.

Процес детекције *outlier*-а дефинисањем асоцијационих правила може дати квалитетније резултате уколико се претходно дефинише стопа минимума сличности између података, као и уколико креирана правила обухватају већи број атрибута. Вредност минималне стопе сличности може се одредити емиријски. Истраживања су показала да најбоље резултате метод даје уколико је минимална стопа сличности између 98.8% и 99.7%. Овакава минимална стопа сличности даје мање лажно негативних и лажно позитивних резултата. Како би се овакав метод успешно могао применити, најпре се ради на нормализацији података, а затим се врши поређење између сваког пара атрибута у скупу података. Погодност примене оваквог метода заснива се на томе да се целокупан процес реализује једним проласком кроз скуп података. Резултат поређења се памти као низ података у меморији. У другом пролазу креирани низ података се узима из

меморије. За сваку инстанцу података у скупу података, сваки пар атрибута који одговара шаблону се проверава како би се видело да ли су вредности у одабраним пољима у оквиру везе на коју указује шаблон. Уколико нису, свака од вредности и свако од поља се означава као могућа грешка. Након што се заврши анализа сваког од парова атрибута, означава се просечни број могућих грешака. Поља која су више пута означена као поља са грешком у односу на просечни број могућих грешака, означавају се као поља која садрже најученстаније грешке. Примена оваквих метода одликује се малом сложеношћу, што их додатно чини погоднијим од, на пример, кластеризационих метода. Мала сложеност и велика брзина извршења чине их погодним за откривање *outlier*-а у токовима података [75].

4.4 Креирање и обука предикционог модела

Након дефинисања скупа података за обуку и извршене предобраде долази на ред кључан корак избора специфичних алгоритама који се могу користити у процесу креирања и обуке предикционог модела. У зависности од садржаја скупа података, као и од очекиваног резултата предикције различити алгоритми надгледаног учења се могу користити са циљем креирања адекватног предикционог модела. Одређени алгоритми машинског учења, као резултат извршења над скупом података, креирају обучени модел спреман за даљу евалуацију и употребу. Са друге стране, постоје и алгоритми чији резултат извршења можда неће бити адекватан предикциони модел, али они свакако доприносе анализи података, утврђивању међусобне зависности између података и креирању структуре погодније за даљу обраду. Група data mining метода погодних за креирање предикционих модела може се поделити на класификационе технике, кластеризационе технике и технике базиране на стаблима одлучивања. Као посебна група могу се издвојити математичке и статистичке методе. Ове методе се првенствено примењују у анализи података, мада се успешно могу применити и као методе за креирање предикционих модела. Неке од значајних математичких и статистичких, класификационих, и кластеризационих метода, као и стабла одлучивања описане су у наставку овог рада.

4.4.1 Математичке и статистичке методе

Математичке и статистичке методе се могу применити у процесу предиктивне анализе, како у домену анализе скупова података тако и у домену креирања и употребе предикционих модела. Линеарна регресија је основни и најчешће коришћени тип предиктивне анализе. Основна идеја математичке регресије је да се најпре провери да ли се на основу скупа предиктивних променљивих може извршити квалитетна предикција зависне променљиве, а затим да се провери које од независних променљивих су од посебног значаја као предиктори зависне променљиве, на који начин су у вези са зависном променљивом и на који начин утичу на предикциони исход. Процене добијене применом регресионих метода користе се у циљу разумевања везе између зависне променљиве и једне или више независних променљивих. Најједноставнија форма регресионе једначине са једном зависном и једном независном променљивом дефинише се једначином:

$$y = \alpha + \beta x$$

У датој једначини y представља вредност зависне променљиве, α константу, β регресиони коефицијент, а x вредност независне променљиве. Овако дефинисана регресиона анализа може се применити у три основне сиуације. Прва примена се огледа у идентификовању ефекта који независна променљива или независне променљиве имају на зависну променљиву. Друга примена је предвиђање ефекта или утицаја могућих промена. Регресиона анализа може показати колико ће се вредност зависне променљиве променити са одређеним степеном промене једне или више независних променљивих. Трећа примена карактерише се предикцијом трендова или будућих вредности. У овом случају регресиона анализа се може користити у процени крајње вредности. На основу свега наведеног може се издвојити неколико типова линеарне регресионе анализе који се могу применити на различите скупове података.

*4.4.1.1 Проста линеарна регресија (***енг. *Simple linear regression)***

Проста линеарна регресија се карактерише једноставним односом између једне зависне и једне независне променљиве. Релација између ове две променљиве је креирана тако да независна променљива што је могуће тачније врши оцену вредности зависне променљиве. Линеарна функција којом се описује однос

између независне и зависне променљиве је невертикална права линија док се независна и зависна променљива могу представити као тачке у дводимензионалном картезијевом координатном систему. Назив проста потиче од чињенице да вредност излазне (зависне) променљиве зависи само од вредности једне независне променљиве. Уколико се посматра дата једначина линеарне регресије као модел може се уочити да дати однос независне и зависне променљиве није одржив када се ради са великим скуповима вредности зависних и независних променљивих. Овакве девијације називијају се грешкама. Из тог разлога могуће грешке се такође морају укључити у сами модел. Уколико се посматра n парова података $\{(x_i, y_i), i = 1, \dots, n\}$, једначина модела линеарне регресије са укљученим параметром грешке може се представити као:

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

Параметар грешке у датој једначине означен је са ε_i. Управо однос између тачних вредности параметара α и β, као података са којима се ради, назива се моделом просте линеарне регресије. Циљ је пронаћи процењене вредности параметара $\hat{\alpha}$ и $\hat{\beta}$ за параметре α и β који ће довести до најбољег уклапања у погледу података. Процена најбољег укпалња врши се методом најмањих квадрата, што се може посматрати као линија која минимизује суму квадратних резидуа $\hat{\varepsilon}_i$ (разлика између стварне и предикцијом добијене вредности зависне променљиве *y*), од којих је сваки дат за све вредности параметара кандидата a и b.

$$\hat{\varepsilon}_i = y_i - a - bx_i$$

Другим речима, $\hat{\alpha}$ и $\hat{\beta}$ решавају следећи минимизациони проблем:

$$Find \ \min_{a,b} Q(a, b),$$

$$Q(a, b) = \sum_{i=1}^{n} {\hat{\varepsilon}_i}^2 = \sum_{i=1}^{n} (y_i - a - bx_i)^2$$

Проширењем до квадратног израза у a и b, могу се добити вредности a и b које минимизују функцију Q. Оне минимизиране вредности су означене са $\hat{\alpha}$ и $\hat{\beta}$ [76].

$$\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x},$$

$$\hat{\beta} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{Cov(x,y)}{Var(x)} = r_{xy}\frac{s_y}{s_x}$$

У датој једначини $\bar{x}$ и $\bar{y}$ представљају просечну вредност x_i и y_i вредности респективно, r_{xy} представља прости корелациони коефицијент између *x* и *y*. s_x и s_y представљају некориговану стандардну девијацију вредности *x* и *y*, док *Var* и *Cov* представљају варијансу и коваријансу респективно.

Заменом горњих израза за $\hat{\alpha}$ и $\hat{\beta}$ у следећој једначини

$$f = \hat{\alpha} + \hat{\beta}x,$$

добија се израз:

$$\frac{f - \bar{y}}{s_y} = r_{xy}\frac{x - \bar{x}}{s_x}$$

Последњи израз показује да је r_{xy} нагиб регресионе праве стандардизованих тачака података и да линија пролази кроз координатни почетак. Генерализовањем $\bar{x}$ нотације може се креирати просечна вредност израза над скупом податка као:

$$\overline{xy} = \frac{1}{n}\sum_{i=1}^{n} x_i y_i$$

Оваква нотација омогућава креирање концизније формиле за нагиб регресионе праве:

$$r_{xy} = \frac{\overline{xy} - \bar{x}\bar{y}}{\sqrt{\left(\overline{x^2} - \bar{x}^2\right)\left(\overline{y^2} - \bar{y}^2\right)}}$$

Степен одређености (R^2) је једнак r_{xy}^2 када је модел линеаран са једном независном променљивом. Такође, овако креиран коефицијент детерминације R^2 се може користити као једна од оцена квалитета креираног линеарног модела.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Ако је модел добар, онда је вредност суме у броиоцу мала, те је вредност за R^2 близу јединице.

Најпознатија мера линеарне корелације између случајних променљивих је *Pearsonov* коефицијент линеарне корелације. Вредност коефицијента корелације креће се у интервалу $[-1,1]$. У складу са величином овог коефицијента, може се закључити смер и интензитет линеарне корелације међу посматраним величинама. *Pearsonov* коефицијент је бездимензионална величина која се може рачунати и по формули:

$$r = \frac{cov(X,Y)}{S_{nx}S_{ny}}$$

Поред процене тачности креираног регресионог модела применом метода најмањих квадрата и *Pearsonov-og* коефицијента, у употреби су и друге регресионе методе. Неке од њих су најмање апсолутно одступање, као и *Theil-Sen* метод који врши одабир линије чији је нагиб средина нагиба одређеног паром узорака података.

*4.4.1.2 Вишеструка линеарна регресија (***енг. *Multiple Linear Regression)***

Како би се описала вишеструка линеарна регресија, принцип линеарне регресије се може описати на другачији начин. Конструкција општег линеарног модела за случајни скуп корелисаних посматрања (инстанци података) заснована је на вектору зависних променљивих y_{NxK} који је повезан са вектором β_{Kx1} који садржи K параметара помоћу познате матрице неслучајних вредности X_{NxK} и вектором случајних вредности грешке ε_{Nx1}. Средња вредност вектора грешке једнака је нули, $E(\varepsilon) = 0$. Матрица коваријансе означена је са $\Omega = cov(\varepsilon)$. На основу свега наведеног репрезентација општег линеарног модела дата је као:

$$y_{Nx1} = X_{NxK}\beta_{Kx1} + \varepsilon_{Nx1}$$

$$E(\varepsilon) = 0 \; i \; cov(\varepsilon) = \Omega$$

На основу свега наведеног може се претпоставити да је сваки елемент y_i вектора y_N у вези са k линеарно независних променљивих.

$$y_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{ik}\beta_k + \varepsilon_i$$

За свако $i=1,2,\ldots,n$ релација између зависне променљиве y и k независних променљивих $x_1, x_2, \ldots, x_k$ је линеарна по параметрима, уколико параметри $\beta_0, \beta_1, \ldots, \beta_k$ варирају над целим простором параметара тако да не постоје ограничења на $\beta_q' = [\beta_0, \beta_1, \ldots, \beta_k]$ где је $q=k+1$ и да вредности грешке ε_i имају заједничку нулу и непознату варијансу σ^2. Уколико је $N=n$ и $K=q=k+1$ може се креирати универијабилни (линеарни) регресиони модел:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{12} & x_{22} & \ldots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1n} & x_{nk} & \ldots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

$$y_{nx1} = X_{nxq}\beta_{qx1} + \varepsilon_{nx1}$$

$$cov(y) = \sigma^2 I_n$$

Где матрица X има потпуни ранг матрице колона, $r(X)=q$. Уколико је $r(X)<q$, тако да X није потпуни ранг матрице, колона и X садржи индикатор променљиве, добија се модел анализе варијанце (*ANOVA*). Веома често матрица X је подељена у два сета независних променљивих који чине матрица A_{nxq_1}која није потпуни ранг и матрица Z_{nxq_2}која је путпуни ранг, тако да матрица X добија следећи облик $X = [AZ]$, где је $q = q_1 + q_2$. У оваквом облику матрица A представља *ANOVA* матрицу, док матрица Z представља регресиону матрицу, која се још назива и матрица коваријабиле. За $N=n$ и $X = [AZ]$ горе наведени универијабилни регресиони модел се назива *ANCOVA* модел. Уколико $\beta' = [\alpha'\gamma']$, *ANCOVA* модел има следећу општу линеарну форму:

$$y = [AZ]\begin{bmatrix} \alpha \\ \gamma \end{bmatrix} + \varepsilon$$

$$y = A\alpha + Z\gamma + \varepsilon$$

$$cov(y) = \sigma^2 I_n$$

Коришћење вишеструке линеарне регресије примењује се у процесу одређивања најадекватнијег линеарног модела за предикцију једне зависне променљиве у на основу фиксног скупа посматраних независних променљивих $x_1, x_2, \dots, x_k$, при чему се независне променљиве узимају без могућности појаве грешке. Употреба универијабилног модела дефинисаног изнад, уз помоћ методе најмањих квадрата, може довести до процене непознате варијансе σ^2.

Како би се дата хипотеза тестирала претпоставља се да зависна променљива *y* прати мултиваријабилну нормалну расподелу са коваријансом $\Omega = \sigma^2 I_n$. Како би се иницијални модел прилагодио подацима, подешавање модела је неизбежан процес у регресионој анализи. Овај процес укључује евалуацију претпоставки модела мултиваријабилне нормалности, хомогеност варијансе и назависност. Уколико су све претпоставке тачне, добија се модел који има најбоље поклапање. На крају, као додатна потврда, може се вршити евалуација предикционог модела применом модела валидације.

Формални тестови и визуелизација података омогућавају систематску помоћ приликом евалуације претпоставки, детекције *outlier*-а, селекције независних променљивих, детектовању инстанци података које имају утицаја на предикцију и детекције недостатка независности [77]. Када се за зависну променљиву *y* може претпоставити да има независну мултиваријабилну нормалну дистрибуцију, али се за структуру коваријансе не може претпоставити сферични облик $\Omega = \sigma^2 I_n$, у таквом случају може се користити генерализована анализа најмањих квадрата. Коришћењем методе генерализованих најмањих квадрата, претпоставља се да постоји општија структура матрице коваријанси. У овом случају две уобичајене форме за коваријансу су $\Omega = \sigma^2 V$, где је *V* познато и не-сингуларно назван је тежински модел најмањих квадрата (*WLS*), и $\Omega = \Sigma$, где је Σ познато и не-сингуларно, назван је генерализовани модел најмањих квадрата (*GLS*).

*4.4.1.3 Мултиваријантна регресија (***енг. *Multivariate Regression)***

Модели мултиваријантне линеарне регресије (*MR*) не користе се само за предикицију у којој фигурира једна зависна променљива, већ се претежно користе за предикцију у којој фигурира већи број зависних променљивих $y_1, y_2, \dots, y_p$.

Два могућа проширења *MR* модела у зависности од скупа независних променљивих су [78]:

- матрица независних променљивих X је иста за све зависне променљиве;
- свака од зависних променљивих је повезана са различитим скупом независних променљивих, тако да је дозвољена употреба p матрица независних променљивих.

Прва могућност *MR* модела је рестриктивнија од друге могућности. Може се сматрати да је прва могућност специјални случај друге могућности. У *MR* моделу за редове у Y или E матрици претпоставља се да су дистрибуирани независно MVN тако да $vec(E) \sim N_{np}(0, \Sigma \otimes I_n)$. Прилагођавање модела $E(Y) = XB$ матрици података Y под MVN расподелом, процена максималне вероватноће *ML* (енг. *Maximum Likelihood*) за B може се изразизи као:

$$\hat{B}_{ML} = (X'X)^{-1}X'Y$$

ML процена је идентична најбољој јединственој ленеарно непристрасној процени (*BLUE*) добијеној коришћењем мултиваријантног општег метода најмањих квадрата заснованог на чињеници да је квадратна норма Еуклидове матрице, $tr[(Y - XB)'(Y - XB)] = \|Y - XB\|^2$ сведена на свим параметарским матрицама B, за фиксно X [79]. За MR модел $Y = XB + E$, матрица параметара B је облика:

$$B = \begin{bmatrix} \beta_0' \\ - - - \\ B_1 \end{bmatrix} = \begin{bmatrix} \beta_{01} & \beta_{02} & \cdots & \beta_{0p} \\ - - - & - - - & - - - & - - - \\ \beta_{11} & \beta_{12} & \cdots & \beta_{1p} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ \beta_{k1} & \beta_{k2} & \cdots & \beta_{kp} \end{bmatrix}$$

где је $q=k+1$ број независних променљивих додељен свакој од зависних променљивих. Вектор β_0' садржи пресеке док матрица B_1 садржи коефицијенте додељене независним променљивама. Овако дефинисана матрица B се назива сирови облик матрице параметара с обзиром да елементи y_{ij} у Y имају општи облик:

$$y_{ij} = \beta_{0j} + \beta_{1j}x_{i1} + \cdots + \beta_{kj}x_{ik} + \varepsilon_{ij},$$

за свако *i=1,2,...,n,* и свако *j=1,2,...,p.*

Како би се креирала девиациона форма *MR* модела средња вредност мора се израчунати према следећој једначини:

$$\bar{x}_j = \sum_{i=1}^{n} \frac{x_{ij}}{n}, \quad j = 1, 2, \dots, k$$

Уз то, мора се формирати вектор девијације резултата према датој једначини.

$$d_{ij} = x_{ij} - \bar{x}_j$$

Након поменутих израчунавања средње вредности и вектора девијације резултата општи облик MR модела постаје:

$$y_{ij} = \beta_{0j} + \sum_{h=1}^{k} \beta_{hj}\bar{x}_h + \sum_{h=1}^{k} \beta_{hj}(x_{ih} - \bar{x}_h) + \varepsilon_{ij}$$

Како би се добила матрична репрезентација горе датог израза могу се увести одређене смене:

$$\alpha_{0j} = \beta_{0j} + \sum_{h=1}^{k} \beta_{hj}\bar{x}_h \quad j = 1,2, \dots p$$

$$\alpha_0' = [\alpha_{01}, \alpha_{02}, \dots, \alpha_{0p}]$$

$$B_1 = [\beta_{hj}] \quad h = 1,2, \dots, k \quad i \quad j = 1,2, \dots, p$$

$$X_d = [d_{ij}] \quad i = 1,2, \dots, n \quad i \quad j = 1,2, \dots, p$$

што доводи до следеће матричне репрезентације:

$$\underset{nxp}{Y} = [1_n X_d] \begin{bmatrix} \alpha_0' \\ B_1 \end{bmatrix} + E$$

Где је 1_n вектор од *n* јединица.

4.4.1.4 Дискриминациона анализа

Технике које припадају групи дискриминационе анализе користе се како би се конкретна променљива класификовала у једну од две или више алтернативних група. Класификација конкретне променљиве врши се на основу низа спроведених мерења. Како је познато, свака од група представља различит скуп података, па самим тим свака променљива припада једном скупу податка. Поред самог процеса класикације променљивих у конкретне групе, ове технике се могу користити како би се одредило које променљиве доприносе процесу класификације. Самим тим, као и у свим data mining техникама и ове технике се према улози могу поделити на предиктивне и дескриптивне. Дискриминациона анализа се може посматрати и као вишеваријантна процедура одређивања значајних променљвих и редуковања скупа функција којима се веза између независних и зависних променљвивих скупа података може описати. Дискриминанте које су линеарне функције променљивих називају се линеарне дискриминативне фунцкије *LDF* (енг. *Linear Discriminant Functions*). Број функција потребних за одржавање максималног раздвајања иницијалног подскупа променљивих назива се ранг или димензионалност раздвајања [74]. Циљеви дискриминационе анализе су конструисање скупа дескриминанти које се могу користити за описивање или карактеризацију раздвајања групе података на основу редукованог скупа променљивих, анализа доприноса оргиналних променљивих на процес раздвајања у оквиру групе и валидација степена раздвања.

Једна од уско повезаних вишеваријантих техника са дискриминационом анализом је и класификациона анализа. Ова анализа креира скуп правила којима се регулише алокација или додељивање инстанци података из скупа података једној или више засебних група. Класификациона правила обично захтевају веће познавање скупа података и параметриску структуру креираних група. Циљ класификационе анализе претежно је смањење укупне вероватноће погрешне класификације података или просечне цене поновне каласификације. С обзиром да се линарна дисриминациона функција користи за креирање класификационих правила циљеви ова два процеса често се преклапају.

Процес дискриминационе анализе заснован на две групе независних узорака од којих се свака састоји од две вишеваријантне нормалне популације података са заједничном матрицом коваријансе Σ и непознатим аритметичим μ_1, μ_2 срединама представља један од најчешће примењиваних метода дискриминационе анализе. Према томе $y_{ij} \sim IN_P(\mu_i, \Sigma)$ где је $i = 1,2;\ j = 1,2, \dots, n_i$ у свако од посматрања y_{ij} је *p x 1* вектор независних променљивих. На основу свега наведеног, може се дефинисати једна од основних дискриминационих анализа под називом *Fisher*-ова дискриминациона анализа [80]. *Fisher*-ова линеарна дискриминациона функција *L* је линеарна комбинација променљивих које обезбеђују максимално раздвајање између група података, која се може представити као:

$$L = a'y = \sum_{j=1}^{p} a_j y_j$$

Вектор које обезбеђује максимално раздвајање између дискриминативних резултата $L_{ij} = a_s' y_{ij}$ за ситуацију са две групе може се дефинисати као:

$$a_s = S^{-1}(\bar{y}_1 - \bar{y}_2),$$

где је $\bar{y}_i$ средња вредност узорка за посматрање у групи *i* (*i=1,2*) и *S* је независна оцена матрице коваријансе Σ. Разлика између средњих вредности дискриминационог резултата добијена евалуацијом дискриминативних резултатата вектора средњих вредности групе $\bar{y}_i$ представља *Mahalanobis*-ову статистичку вредност D^2:

$$D^2 = \bar{L}_1 - \bar{L}_2 = a_s'\bar{y}_1 - a_s'\bar{y}_2 = a_s'(\bar{y}_1 - \bar{y}_2) = (\bar{y}_1 - \bar{y}_2)'S^{-1}(\bar{y}_1 - \bar{y}_2)$$

Наиме, ако је $T^2 = (\frac{n_1 n_2}{n_1} + n_2)D^2$ значајна тада постоји добро раздвајање центроида за две популације података које се посматрају. Такође, веома је лако утврдити да је квадрат универијабилне студентове t^2 статистике коришћењем средње вредности дискриминативних резултата за две групе једнак T^2 с обзиром да је:

$$t^2 = \frac{(\bar{L}_1 - \bar{L}_2)^2}{s_L^2\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} = \left(\frac{n_1 n_2}{n_1 + n_2}\right)(\bar{y}_1 - \bar{y}_2)' S^{-1}(\bar{y}_1 - \bar{y}_2) = T^2,$$

где је s_L^2 процена заједничке варијане популације за дискриминативне резултате.

Уколико се постави да дискриминативни резултати $L_{ij} \equiv y$ представљају зависну променљиву и да су независне променљиве за две групе $x_1 = -1$ и $x_2 = 1$, може се уклопити регресиона једначина $y = \alpha + \beta x + \varepsilon$ у дискриминативне резултате. Тестирање $H: \beta = 0$ у оквиру регресионог проблема је еквивалентно тестирању $\rho = 0$ коришћењем t^2 статистике.

$$t^2 = \frac{(n-2)r^2}{1 - r^2}$$

Уколико се израз за t^2 замени у претходом изразу добија се следећа једначина:

$$\frac{n_1 n_2 D^2}{n_1 + n_2} = \frac{(n-2)r^2}{1 - r^2}$$

$$D^2 = \frac{(n_1 - n_2)(n-2)r^2}{(n_1 n_2)(1 - r^2)}$$

$$r^2 = \frac{n_1 n_2 D^2}{(n_1 + n_2)(n-2) + n_1 n_2 D^2}$$

Где је $n = n_1 + n_2$. Извођењем псеудорегресије над независним променљивама

$$c_1 = \frac{n_2}{n_1 + n_2} \quad , \quad c_2 = c_1 - 1 = -\frac{n_1}{n_1 + n_2}$$

за групе 1 и 2 респективно, могу се регресирати оригиналне променљиве на вештачке променљиве. Након тога уколико се r^2 замени са R_p^2, добија се квадратна вишеструка корелација или одређивање коефицијената регресијом незвисне променљиве c_i на зависне променљиве $y_1, y_2, \dots, y_p$. Уколико је вектор $b' = [b_1, b_2, \dots, b_p]$ вектор процењених регресионих коефицијената, тада је вектор b пропорцијалан вектору a_s.

$$b = \left[\frac{n_1 n_2}{(n_1 + n_2)(n_1+n_2 - 2) + n_1 n_2 D_p^2}\right] a_s$$

$$a_s = \left[\frac{(n_1 + n_2)(n_1+n_2 - 2)}{n_1 n_2} + D_p^2\right] b$$

Добија се и следећа једнакост $R_p^2 = (\bar{y}_1 - \bar{y}_2)'b$ [81]. На овакав начин могу се добити вредности за b и R^2 из вектора a и D_p^2.

4.4.2 Класификационе data mining технике

Класификација је један од најчешћих задатка машинског учења и представља проблем разврставања непознате инстанце у једну од унапред понуђених категорија или такозваних класа. Принцип на коме се базира рад класификационих data mining техника омогућио је примену истих на решавање најразличитијих проблема. Неки од њих су дијагностификовање болести, прогнозе болести код пацијената, класификација кредитних захтева клијената, анализа слике и гласа за биометријске потребе, дијагностификовање здравственог стања биљака или животиња и слично. Циљна функција проблема класификације је дискретна, што значи да се у општем случају ознакама класа не могу смислено доделити нумеричке вредности нити уређење, тако да је класни атрибут, чију је вредност потребно одредити, заправо категорички атрибут. Класификација неког објекта се заснива на проналажењу сличности са унапред одређеним објектима који су припадници различитих класа, при чему се сличност два објекта одређује анализом њихових карактеристика. При класификацији се сваки објекат сврстава у неку од класа са одређеном тачношћу. Задатак је да се на основу карактеристика објеката чија класификација је унапред позната, направи модел на основу кога ће се вршити класификација нових објеката. У проблему класификација, број класа је унапред познат и ограничен. Процес класификације се састоји из две фазе [9]. Као и код раније поменутих процеса, рецимо кластеризације, у првој фази се гради модел на основу карактеристика објеката чија класификација је позната. За изградњу модела се користе подаци који се најчешће налазе у табелама. Свака инстанца узима само једну вредност атрибута класе, а атрибут класе може да има коначан број дискретних вредности које нису уређене.

Класификациони алгоритам учи на основу познатих класификација тј. на основу инстанци објеката чија класификације је позната. При томе, на основу вредности њихових атрибута и атрибута класе, гради се скуп правила на основу којих ће се касније вршити класификација. Након учења, модел се тестира, при чему под тачношћу подразумевамо проценат инстанци које су тачно класификоване. Вредност атрибута класе сваке инстанце из тест скупа података пореди се са вредношћу атрибута класе која је одређена на основу модела. Важно је напоменути да се за тестирање модела користе инстанце које нису коришћене у фази учења. Постоји више начина за издвајање тестних инстанци, али се најчешће издвајају случајним избором, пре фазе учења, од инстанци чија је класификација позната. При томе, ако је тачност модела задовољавајућа, онда се даље користи у класификацији објеката чија вредност атрибута класе није позната.

Широк распон алгоритама за класификацију је на располагању, сваки са својим предностима и недостацима. Оно што је сигурно, јесте да не постоји алгоритам надгледаног машинског учења који даје најбоље резултате у раду са свим проблемима.

4.4.2.1 К-најближих суседа

Алгоритам класификације под називом *к*-најближих суседа припада групи класификационих модела чији је рад заснован на раду са инстанцама података. Овај алгоритам је коришћен још током педесетих година двадестог века, док је најинтезивније коришћење било на распознавању узорака. Потребно је нагласити да класификација на инстанцама спада у једне од најједноставниих техника анализе података. Разлог једноставности лежи у томе што се не врши експлицитна генерализација циљног појма на основу особина које се могу добити на основу скупа за обуку модела, већ се своди на памћење тренинг скупа, односно појединачних инстанци које скуп података садржи. Ово значи да основни облик алгоритма *к*-најближих суседа не укључује процесирање инстанци из скупа за учење у фази конструкције модела, већ само њихово меморисање. Процес класификације нових инстанци заснован је на поређењу нове инстанце са меморисаним инстанцама из тренинг скупа податкаа коришћењем унапред дефинисане метрике. Метрика конкретно дефинише растојање инстанци базирано

на основу вредности њихових атрибута. Уколико су вредности атрибута у инстанцама сличније, растојање је мање. Самим тим, нова инстанца се класификује на основу претраживања скупа података при чему се проналази инстанца која је по растојању најближа новој инстанци. Након проналаска најближе инстанце, новој инстанци се додељује класа пронађене најближе инстанце. Као метрика растојања најчешће се код data mining технке *к*-најближих суседа користи Еуклидово растојање које се може дефинисати као:

$$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

При чему је са $x = x_1, x_2, \dots, x_n$ представљен вектор вредности атрибута произвољне инстанце података.

Друга репрезентација Еуклидовог растојања између инстанци података *x* и *y* може се представити као:

$$d(x, y) = \|x - y\| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Друге метрике се могу дефинисати и варирањем потенције координата вектора у дефиницији Еуклидове метрике. На пример уколико се изостави квадрат и употреби апсолутна вредност, добија се такозвана правоугаона метрика. Нереално велике разике у вредностима појединих координата додатно се наглашавају вишим потнецијалима, науштрб координата код којих је разлика у вредности мала.

Када се ради са нумеричким атрибутима у оквиру инстанци података често долази до проблема везаног за коришћење различитих метрика. Тако, уколико нумерички атрибут има распон вредности унутар десетих делова мерне јединице, такав атрибут неће имати великог утицаја на коначан резултат у односу на неке друге атрибуте чије су вредности са распоном од неколико десетина мерних једница. Управо из ових разлога врло често се захтева нормализација података са

којима мотоди класификације засновани на инстанцама раде. Још једна од потенцијалних модификација почетног скупа података извршава се када се у скупу података, поред нумеричких вредности, налазе атрибути који имају номиналне вредности. Како би се над номиналним вредностима применила дефиниција Еуклидовог растојања потребно је дефинисати операцију разлике над номиналним вредностима. С тим у вези, ако су $a_i, a_j \in Dom(A_i)$ две произвољне вредности номиналног атрибута A_i, онда се разлика вредности a_i и a_j може дефинисати 0-1 функцијом разликовања, на следећи начин:

$$a_i - a_j = \begin{cases} 0, za\ a_i = a_j \\ 1, za\ a_i \neq a_j \end{cases}$$

Општа дефиниција Еуклидовог растојања дата у изразима изнад може се користити и за инстанце с неодређеним вредностима атрибута, уколико се овако дефинисана функција разлике примени над датим подацима. Код класификације засноване на инстанцама не спроводи се експлицитна генерализација својстава циљног појма, тј. не претражује се простор решења у потрази за што бољим моделом. Код класификације се појављује само један имплицитни класификацијски модел који је у потпуности одређен скупом за учење и функцијом растојања. Основни алгоритам се може надограђивати тако да се у одређеном опсегу модификује скуп инстанци за учење или функција растојања. Поред модификације скупа инстанци које ће се памтити и на основу којих ће се вршити обука модела на облик и перформансе класификационог модела може се утицати и модификацијом функције растојања. Функција растојања базирана на Еуклидовој метрици заснива се на претпоставци да је утицај свих атрибута у инстанци на коначан резултат једнак. Оваква ситуација је у пракси јако ретка, па се самим тим отвара могућност побољшања технике класификације. Модификацијом функције растојања на такав начин да њен модификован облик валоризује класификацијски потенцијал различитих атрибута једно је од могућих решења. Ово се може постићи проширењем Еуклидовог растојања увођењем, примера ради, тежинских вредности атрибута. Овако придружена тежинска вредност атрибуту A_i може се означити са w_i. На основу уведених тежинских вредности Еуклидово растојање инстанци *x* и *y* може се дефисати као:

$$d_w(x,y) = \sqrt{\sum_{i=1}^{n} w_i^2 (x_i - y_i)^2}$$

Већа тежинска вредност придружена атрибуту пружа већи утицај на израчунавање растојања инстанци. У поменутом процесу свим атрибутима се на почетку придружује тежинска вредност 1, која се итеративно мења при посматрању сваке од инстанци из скупа података. Ако инстанце *x* и *y* припадају истој класи, смањује се тежинска вредност атрибута чије се вредности у инстанцама x и y највише разликују, јер се разлика у вредностима тих атрибута приписује слабијој корелацији са класом, као и што се у случају да инстанце x и y припадају различитим класама, тежинска вредност атрибута са највећом разликом вредности повећава. Повећање, односно смањење тежинске вредности пропорционално је разлици вредности атрибута у инстанцама x и y.

Промена вредности параметра k може довести до непостојаности класификације. Вредност параметра k се најчешће одређује емпиријски, евалуацијом класификације за различите вредности самог параметра k, при чему се бира вредност k за коју је класификација била најуспешнија. Методе класификације засноване на инстанцама не граде експлицитан модел података у виду неке функције као што то ради већина метода машинског учења. Зато се класификација не врши на основу већ формулисаног модела, него на основу скупа инстанци за тренинг, тако што се инстанце предвиђене за тренирање чувају и бивају употребљене тек кад је потребно класификовати непознату инстанцу. На овај начин се већина израчунавања премешта из фазе обуке модела у фазу примене. На основу свега наведеног, основна предност класификације засноване на инстанцама је могућност истраживања произвољних по деловима линеарних граница међу класама. Са друге стране, основни недостатак је чињеница да класификацијски модел није изражен експлицитно, односно у облику који би био дескриптиван у терминима домена класификацијског проблема.

4.4.2.2 Naive Bayes класификатор

Основна компонента *Naive Bayes* класификатора је *Bayes*-ова теорема. Уједно ова теорема је и основ пробабилистичког приступа који додељује свакој инстанци

вероватноћу класификације у одређену класу. Уколико се процес класификације претпостави као проналажење највероватније класификације U_{MAP}, тада се сама класификација може израчунати као:

$$v_{MAP} = \underset{v_j \in V}{\operatorname{agrmax}} P(v_j | a_1, a_2, \dots, a_n)$$

При чему је V коначни скуп свих могућих класификација улазне инстанце, а v_{MAP} највероватнији елемент датог коначног скупа. Уколико се свака инстанца прикаже као скуп вредности атрибута и уколико је познат тренинг скуп дефинисан такође истим скупом атрибута, претходни израз се може другачије представити као:

$$v_{MAP} = \underset{v_j \in V}{\operatorname{agrmax}} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)}$$

$$v_{MAP} = agr\max_{v_j \in V} P(a_1, a_2, \dots, a_n | v_j) P(v_j)$$

Проблем који се може јавити са оваквом репрезентацијом произилази из међусобне зависности вредности атрибута тако да је број могућих израза једнак броју свих могућих различитих n-торки $\{a_1, a_2, \dots, a_n\}$ помножних са бројем свих могућих класификација. Овако дефинисан класификатор уводи поједностављење у виду претпостављене међусобне независности вредности арибута у n-торкама $\{a_1, a_2, \dots, a_n\}$. Увођењем дате претпоставке, претходни израз се може написати као:

$$P(a_1, a_2, \dots, a_n | v_j) = \prod_i P(a_i | v_j)$$

Све преходно наведено доводи до коначног израза за *Naive Bayes* класификатор:

$$v_{NB_j} = agr \max_{v_j \in V} P(v_j) \prod_i P(a_i | v_j)$$

Број различитих вероватноћа које треба израчунати из датих података у оквиру тренинг скупа износи број различитих вредности атрибута помножен са бројем различитих могућих класификација, што представља значајно мањи број него број потребан како би се добила вероватноћа $P(a_1, a_2, \dots, a_n|v_j)$.

Теоретски посматрано претпостављен услов независности може бити релативно строг и може представљати проблем. Ипак, практична реализација, а самим тим и употреба *Naive Bayes* класификатора показала се веома практичном због једноставне имплементације и добрих класификационих резултата које остварује. Такође, још једна од предности употребе овако дефинисаног класификатора је и мала захтевност меморије, као и то што је робустан за нерелевантне податке, јер се овакви подаци могу поништавати. Са друге стране, овај класификатор се добро показао и у раду са скуповима података који садрже велики број подједнако релевантних података. Оптималност овог класификатора огледа се у домену тачне претпоставке независности података.

*4.4.2.3 SVM (***енг.** *Support Vector Machine)*

SVM метода у основи представља бинарни класификатор којим се на основу конструисане хипер равни у високодимензионалном простору врши предикција припадности нове инстанце података једној од две могуће класе. Ова метода, такође, спада у методе из групе надгледаног машинског учења. Основа идеја ове методе је дефинасана на статистичким теоријама, те самим тим смањује могућност погрешне класификације. Идеја је базирана на томе да се пронађе раздвајајућа хипер раван тако да сви подаци из дате класе буду са исте стране креиране равни. *SVM* метода одређује оптимално решење које максимизује раздаљину између хипер равни и тачака које су близу потенцијалне линије раздвајања. Тако, уколико нема тачака близу линије раздвајња, онда ће класификација бити прилично лака. Примена SVM метода може се разликовати у зависности од тога да ли се примењује на линеарно нераздвајајуће или линеарно раздвајајуће проблеме. У случају линеарно не-раздвајајућих проблема, користи се нелинеарни *SVM*, при чему је основна идеја да се основни (улазни) векторски простор преслика у неки вишедимензиони простор у коме је скуп података за тренинг линеарно раздвојив.

Уколико у скупу података, на основу којих се обавља тренинг модела, постоји *L* вектора, односно тачака у *D*-димензионалном простору, при чему сваки узорак x_i има *D* атрибута, односно компоненти вектора и припада једној од две класе y_i=-*1* или y_i=*1*, тада се облик једног улазног податка може представити следећим изразом:

$$\{x_i, y_i\}\ gde\ je\ i = 1 \dots L, y_i \in \{-1,1\} x \in R^D$$

На основу наведеног се претпоставља да су подаци линеарно одвојиви, што значи да се може нацртати правац у координатном систему за случај *D=2*, при чему су осе означене са x_1 и x_2, односно хипер раван за случај *D>2*. Овако креирана хипер раван може се описати изразом:

$$w \cdot x + b = 0,$$

при чему је *w* нормала хипер равни, а $\frac{b}{\|w\|}$ представља вертикалну удаљеност хипер равни од координатног почетка система. Тачке које су најближе хипер равни су *suport* вектори који се најтеже класификују. Основни циљ SVM метода је одабир хипер равни која је максимално удаљена од најближих инстанци обе класе. С обзиром да је избор параметара *w* и *b* кључан, имплементација SVM методе се своди на избор истих, при чему се улазни подаци могу описати на следећи начин:

$$x_i \cdot w + b \geq +1\ za\ y_i = +1$$

$$x_i \cdot w + b \leq -1\ za\ y_i = -1$$

Уколико се изврши комбиновање претходна два израза добија се:

$$y_i(x_i \cdot w + b) - 1 \geq 0, \forall i$$

На овакав начин равни H_1 и H_2 на којима леже SVM вектори могу се приказати помоћу следећих израза:

$$x_i \cdot w + b = +1\ za\ H_1$$

$$x_i \cdot w + b = -1\ za\ H_2$$

Растојање равни H_1 и H_2 до хипер равни може се дефинисати помоћу вредности d_1 и d_2, при чему еквидистантност хипер равни од H_1 и H_2 подразумева $d_1 = d_2 = \frac{1}{\|w\|}$. Вредности растојања d_1 и d_2 називају се маргином. Одабир хипер равни која је максимлно удаљена од вектора своди се на максимизирање вредности за маргину, што одговара проналажењу: $min\|w\|$

Takve da je $[\![y_i(x]\!]_i \cdot w + b) - 1, za \ \geq 0, \forall i$

Када је реч о линеарно нераздвајајућим класама, како би се користиле *SVM* методе, долази до увођења негативне вредности ξ_i. На основу уведене негативне вредности могу се дефинисати следећи изрази:

$$x_i \cdot w + b \geq +1 - \xi_i \, za \, y_i = +1$$

$$x_i \cdot w + b \leq -1 + \xi_i \, za \, y_i = -1$$

Уколико се, као и у претходном случају када је било речи о линеарно раздвојивим класама, изврши комбинација последња два израза добија се:

$$[\![y_i(x]\!]_i \cdot w + b) - 1 + \xi_i \geq 0, \xi_i \geq 0, \forall i$$

Описана метода се за разлику од методе маргине назива методом меке маргине јер дозвољава погрешно означавање класа пре самог почетка тренинг процеса. У овом случају мера растојања инстанце од припадајућег *SVM* вектора означава се са ξ. На основу свега наведеног избор хипер равни своди се на проналажење:

$$\frac{min1}{2}\|w\|^2 + C\sum_{i=1}^{L}\xi_i$$

Такав да $[\![y_i(x]\!]_i \cdot w + b) - 1 + \xi_i \geq 0, \forall i$

У датом изразу вредност C представља фактор грешке. Фактор грешке представља дозвољену стопу грешке при тренинг процесу и дефинисан је унапред пре почетка тренинг процеса. Исто тако проналазак хипер равни не би био могућ без увођења фактора грешке [9].

Када се ради о комплексноти ове методе иако обука модела применом SVM методе може бити захтевна за велики број примера и класа она је генерално линеарно комплекса *O(nm),* при чему је m димензија простора. Уколико се упореди са сличним методама машинског учења, може се закључити да је мање сложености, с обзиром да друге сличне методе експоненцијално зависе од m.

SVM метода може се посматрати као уопштени класификатор оптималне границе за нелинеарну класификацију. Овакав класификатор се добија применом поступка који је познат под називом *Kernel* трик. Како би се добио поменути поступак улазни вектор x_i мора се заменити функцијом $\Phi(x_i)$. Дата функција врши пресликавање n-димензионалног у m-димензионални простор, уз $m \gg n$, како би добили узорке који су ленеарно раздвојиви. Како би се израчунао производ вектора $\Phi(x_i)$ и w врши се одабир *Kernel* функције $K(x_{i,} \quad x_j) = \Phi(x_i)^T \cdot \Phi(x_j)$ помоћу које је могуће извршити израчунавање, при чему не долази до проблема са димензионалношћу. Поред *kernel* функције, као што је раније наведено, фактор грешке је још један од битнијих параметра. Када се ради о одабиру функције *kernel*-а, свака функција која задовољава услове симетричности, позитивности и дефинисаности може се применити као *kernel* функција. Како би одабрна *kernel* функција била одговарајућа, потребан и довољан услов је да задовољава *Mercer*-ову теорему.

У практичној имплементацији SVM *kernel*-а користе се неколико стандардизованих облика *Kernel* функције: линеарна, полиномна, Гаусова, рационална квадратна исигмоидална. Линеарни *kernel* се подједнако добро може применити, како код линеарно раздвојивих података тако и случајевима нелинеарних података. Перформансе полиномног *kernel*-а требало би да буду отприлике исте као код линеарног *kernel*-а. Са друге стране, Гаусов *kernel* је доста бољи од линеарног или полиномног *kernel*-а. Међутим, Gausov *kernel* је тешко имплементирати због комплексног одабира одговарајућег фактора грешке. И поред сложене имплементације, Гаусов *kernel* има најширу примену која одговара пресликавању у бесконачно-димензионални простор. Сигмоидни *kernel* није ефикасан као што је то случај са осталим *kernel*-има. Овај *kernel* не мора да буде позитивно дефинисан.

Према општој дефиницији SVM метода, може се закључити да је ова метода применљива у случају када се ради са две класе, с обзиром да се ради о бинарном класификатору. Проблем који се јавља у пракси јесте класификација података у више од две класе. Поменути проблем не може се решити применом само једног класификатора, већ се може решити на један од два начина:

- Први начин у оригиналу носи назив *one-versuss-all* и базира се на конструисању *n* бинарних класификатора од којих сваки разврстава узорке или у једну од две класе или у преосталих *n-1* класа. Свака нова инстанца података се класификује коришћењем принципа *winner-takes-all*. Примена ове стратегије подразумева да сваки класификатор поред излаза из процеса класификације даје и меру сигурности у избор класе за дату инстанцу података. Од свих класификатора чији избор није „all“ узима се избор онога који је дао највећу меру сигурности за свој избор класе. Уколико пак сви класификатори одаберу „all“, у том случају постоји вероватноћа да се ради о непостојећој класи података, или са друге стране није могуће класификовати дату инстанцу података. Уколико дође до ситуације да сви класификатори изаберу „all“, најчешће се узима избор супротан оном класификатору који је дефинисао најмању сигурност за своје одабрано „all“.
- Други начин у оригиналу носи назив *one-versuss-one* и базира се на конструкцији $\frac{n(n-1)}{2}$ бинарних класификатора од којих сваки сврстава узорке у једну две класе. Класификација нових инстанци података се врши поступком гласања. Извршена бинарна класификација се сматра једним гласањем за једну од две класе, чиме се број гласова за класу која је одабрана при тој бинарној класификацији увећава за један. Након извршења свих поступака гласања, инстанци података се додељује класа са највише освојених гласова. Уколико између две класе дође до подударања у броју гласова пракса је да се врши одабир класе са мањим индексом.

4.4.2.4 Стабла одлучивања

Стабла одлучивања представљају класификатор изражен у виду рекурзивне поделе простора инстанци. Обука стабла одлучивања представља методу апроксимације дискретних циљних функција у коме се научена функција представља у виду стабла, где сваком чвору стабла одговара тест неког атрибута инстанце, гране које излазе из чвора одговарају различитим вредностима атрибута, а листовима одговарају вредности циљне функције. Стабло одлучивања се састоји од чворова који формирају рутирано стабло. Рутирано стабло представља директно стабло са почетним чвором названим *root* чвор који нема долазне потеге. Сви остали чворови имају тачно један долазни потег. Чворови са излазним потезима назива се интернални (унутрашњи) чвор или тест чвор. Сви преостали чворови у стаблу одлучивања називају се листови, терминални чворови или чворови одлуке. У стаблу одлучивања сваки унутрашњи чвор дели простор инстанци на два или више подпростора према одређеној дискретној функцији улазних вредности атрибута. У најједноставнијем и најчешћем случају сваки тест чвор представља један атрибут, тако да се на такав начин простор инстанци дели у складу са вредностима тог атрибута. Уколико се ради о нумеричким атрибутима, услов дељења се дефинише у одређеном опсегу.

Сваки лист у стаблу одлучивања је додељен једној класи. На овакав начин лист чвор представља одговарајућу циљну вредност. Додатно лист у стаблу одлучивања може садржати вектор вероватноће који показује вероватноћу да циљни атрибут (класни атрибут) има одговарајућу вредност. Инстанце података се класификују методом наниже, почевши од *root* чвора према листовима, у зависности од резултата провере услова гранања на сваком од чворова на датој путањи. Стабла одлучивања могу инкорпорирати номиналне и нумеричке атрибуте. У случају нумеричких атрибута, стабло одлучивања се може геометријски интерпретирати као скуп хипер равни, од којих је свака ортогонална једној од оса [5].

Сложеност стабла одлучивања је кључан ефекат његове тачности, при чему се сложеност стабла може контролисати дефинисањем критеријума завршетка креирања стабла и применом метода одсецања дела стабла (енг. *Pruning*).

Поменута два метода за контролисање сложености стабла одлучивања не искључују један другог. Када се говори о метрици сложености стабла одлучивања користе се следеће метрике: укупан број чворова, укупан број листова, дубина стабла и број искоришћених атрибута. Индуковање стабла одлучивања јако је повезано са индуковањем правила одлучивања. Тачније, свака од путања од *root* чвора стабла одлучивања до једног од листова може се трансформисати у правило одлучивања повезивањем тестова на датој путањи и узимањем предиктивне вредности листа као класне вредности.

Приликом креирања стабла одлучивања веома је битан критеријум поделе на основу кога ће се извршити расподела атрибута у оквиру унутрашњих чворова стабла. У највећем броју случајева дискретна функција поделе је једноваријабилна, што значи да се унутрашњи чворови стабла деле према вредности једног атрибута. Практично, алгоритам на основу кога се креира стабло, такозвани индуктор, врши тражење најбољег атрибута на основу кога ће се извршити подела. Постоје различити једноваријабилни критеријуми и могу се карактеризовати на различите начине:

- Према пореклу мере: теорија информација, зависност и дистанца.
- Према сруктури мере: критеријуми засновани на шуму, нормализовани критеријуми засновани на шуму и бинарни критеријуми.

На основу свега наведеног, могу се издвојити неки од најчешће примењиваних критеријума поделе који се могу наћи у литератури. Критеријум заснован на шуму (нечистоћи) у подацима може се дефинисати као функција $\phi: [0,1]^k \rightarrow R$ која задовољава следеће услове:

- $\phi(\mathrm{P}) \geq 0$;
- $\phi(\mathrm{P})$ је минимум ако постоји i такво да је $p_i = 1$;
- $\phi(\mathrm{P})$ је максимум ако за свако i, $1 \leq i \leq k$, $p_i = {}^{1}/_{k}$;
- $\phi(\mathrm{P})$ је симетрична функција у односу на компоненте вектора P;
- $\phi(\mathrm{P})$ је глатка функција (диференцијабилна) на свом опсегу.

При чему је са *x* означена случајна променљива са *k* дескретних вредности која има расподелу према $P = (p_1, p_2, \dots, p_k)$. Уколико вектор вероватноће има компоненте јединице (променљива *x* добија само једну вредност), тада се таква променљива дефинише као чиста. Са друге стране, уколико су све компоненте једнаке, ниво шума достиже максимум. Уколико се тренинг скуп података означи са *S*, вектор вероватноће селектованог атрибута *y* може се дефинисати као:

$$P_y(S) = \left(\frac{\left|\sigma_{y=c_1} S\right|}{|S|}, \dots, \frac{\left|\sigma_{y=c_{|dom(y)|}} S\right|}{|S|} \right)$$

Доброта расподеле на основу дискретног атрибута a_i дефинисана је као редукција шума селектованог атрибута након дељења *S* на основу вредности $V_{i,j} \in dom(a_i)$ као:

$$\triangle \Phi(a_i, S) = \phi\left(P_y(S)\right) - \sum_{j=1}^{|dom(a_i)|} \frac{\left|\sigma_{a_i=v_{i,j}} S\right|}{|S|} \cdot \phi(P_y(\sigma_{a_i=v_{i,j}} S))$$

Information gain је заправо мера која показује колико добро одређени атрибут врши поделу података у групе на основу дате класификације (класног атрибута). Овај критеријум је заснован на ентропији. Оригинално потиче из теорије информација [82]. Може се дефинисати као:

$$InformationGain(a_i, S)$$
$$= Entropy(y, S) - \sum_{v_{i,j} \in dom(a_i)} \frac{\left|\sigma_{a_i=v_{i,j}} S\right|}{|S|} \cdot Entropy(y, \sigma_{a_i=v_{i,j}} S)$$

Где је:

$$Entropy(y, S) = \sum_{c_j \in dom(y)} -\frac{\left|\sigma_{y=c_j} S\right|}{|S|} \cdot \log_2 \frac{\left|\sigma_{y=c_j} S\right|}{|S|}$$

Gain Ratio као критеријум поделе врши нормализавију критеријума информативне добити на следећи начин [47]:

$$GainRatio(a_i, S) = \frac{InformationGain(a_i, S)}{Entropy(a_i, S)}$$

Овакав однос је недефинисан у случајевима када је именилац једнак нули. Такође, код овог критеријума може доћи до фаворизовања атибута за које је именилац јако мали. Сам процес се спроводи у две фазе. У првој фази израчунава се информативна добит за све атрибуте. За оне атрибут који имају довољно добру просечну информативну добит најбољи *gain ratio* се селектује. Показано је да *gain ratio* тежи да превазиђе једноставне критеријуме за добијање информација како са аспекта тачности, тако и са аспекта сложености класификатора [83].

Мера растојања на сличан начин, као и *gain ratio* врши нормализацију мере шума. Међутим, овај критеријум врши нормализацију на другачији начин [84]. Нормализација помоћу овог критеријума се може приказати следећим изразом:

$$\frac{\triangle \Phi(a_i, S)}{-\sum_{v_{i,j} \in dom(a_i)} \sum_{c_k \in dom(y)} \frac{\left|\sigma_{a_i = v_{i,j} AND y = c_k} S\right|}{|S|} \cdot \log_2 \frac{\sigma_{a_i = v_{i,j} AND y = c_k} S}{|S|}}$$

Бинарни критеријум се користи у процесу креирања бинарних стабала одлучивања. Бинарне мере се базирају на дељењу домена улазних атрибута на два поддомена. Уколико се бинарни критеријум вредности за атрибут a_i над скупом података S означи са $\beta(a_i, dom_1(a_i), dom_2(a_i), S)$, тада су $dom_1(a_i)$ и $dom_2(a_i)$ одговарајући поддомени. Добијена вредност за оптималну поделу домена атрибута у два међусобно искључива поддомена користи се за поређење атрибута.

Процес или фаза раста стабла приликом креирања на основу скупа података мора се ограничити тако што се ће се дефинисати критеријум прекида. Правила за креирање критеријума прекида могу бити различита и сами прекид креирања стабла окида се када је унапред дефинисано правило прекида испуњено. Нека од уобичајених правила прекида дефинисана су на следећи начин:

- све инстанце тренинг скупа података припадају једном излазу *y*;
- максимална дубина стабла одлучивања је достигнута;

- број случајева у терминалном чвору је мањи од минималног броја случајева за родитељски чвор;
- уколико је дошло до дељења чвора, број случајева у једном или више деце чворова биће мањи од минималног броја случајева који могу бити дете чвор;
- најбољи критеријуми раздвајања нису већи од унапред дефинисаног прага.

Примена чврстих критеријума прекида креирања стабла могу довести до креирања малих или подцењених стабала одлучивања. Са друге стране, коришћење лоших критеријума прекида доводи до генерисања великих сабала одлучивања која су подређена целом тренинг скупу података. Још један од начина регулисања величине стабла одлучивања јесте и већ поменуто одсецање стабла. Методе одсецања креиране су како би се решили проблеми одабира одговарајућег критеријума прекида. Према овој методологији се дозвољава креирање што је могуће већег стабла одлучивања применом слабог критеријума прекида, при чему стабло облучивања превазилази тренинг скуп података [85]. Након тога се врши одсецање дела стабла како би се превелико креирно стабло свело на мање. Практично гранања у стаблу одлучивања која недоприносе генерализацији тачности самог стабла се одбацују. Овакав метод може унапредити тачност стабла одлучивања поготову када се ради о подацима са пуно шума. Постоје различите технике за одсецање стабла одлучивања. Већина од њих примењује *top-down* или *bottom-up* обилазак чворова. Конкретан чвор ће бити одсечен уколико сама операција одсецања унапређује одређени критеријум.

Одсецање сложености трошка (енг. *Cost-Complexity Pruning*) једна је од техника одсецања стабла одлучивања. Ова техника се изводи у две етапе. У првој етапи секвенца стабла $T_0, T_1, \dots, T_k$ креирана је над тренинг скупом података, где је T_0 оригинално стабло пре одсецања, а T_k је *root* стабло. У другој етапи, једно од ових стабала се бира као одсечено стабло, на основу којег се врши процена грешке генерализације. T_{i+1} стабло одлучивања добија се заменом једног или више подстабала у стаблу претходника T_i са одговарајућим листовима [85]. Одсечена подстабла су она стабла која добијају најмање повећање стопе грешке по одсеченом листу. Овакво одсецање може се представити изразом:

$$\alpha = \frac{\varepsilon(pruned(T,t),S) - \varepsilon(T,S)}{|leaves(T)| - |leaves(pruned(T,t))|}$$

Где $\varepsilon(T,S)$ представља стопу грешке стабла *T* над узорком података *S*. Такође, $|leaves(T)|$ означава број листова у стаблу, *T*. $pruned(T,t)$ означава стабло добијено заменом чвора *t* у стаблу *T* одговарајућим листом.

У другој фази се врши процена грешке генерализације сваког од одсечених стабала $T_0, T_1, \dots, T_k$. На основу спроведене процене се врши одабир најбољег стабла добијеног одсецањем. Уколико је дати скуп података довољно велики, може се поделити на тренинг скуп и скуп за одсецање. На овакав начин се стабла одлучивања могу конструисати помоћу тренинг скупа, док се евалуација може вршити на основу скупа за одсецање. Са друге стране, уколико скуп података није довољно велики, упркос великој сложености израчунавања, може се користити унакрсна validacija.

Одсецање најмање грешке (енг. *Minimum error pruning*) је још једна од метода одсецања стабла одлучивања [86]. Ова метода користи обилазак стабла од листова према root чвору. Током обиласка у сваком чвору стабла се врши поређење вероватноће грешке са и без одсецања. Процена вероватноће грешке је корекција процене просте вероватноће коришћењем фреквенције. Уколико се са S_t означе инстанце података које су досегле до листа *t*, тада се очекивана стопа грешке у датом листу може представити са:

$$\varepsilon'(t) = 1 - \max_{c_i \in dom(y)} \frac{\left|\sigma_{y=c_i} S_t\right| + l \cdot p_{apr}(y = c_i)}{|S_t| + l}$$

Где је $p_{apr}(y = c_i)$ *a-priori* вероватноћа да ће *y* добити вредност c_i и *l* означава тежину додељену *a-priori* вероватноћи. Стопа грешке унутрашњег чвора представља просечну тежину стопе грешке гранања датог чвора. Тежина се одређује на основу процента инстанци на свакој од грана. Израчунавање се врши рекурзивно до листова. Уколико је унутрашњи чвор одсечен онда он постаје лист и његова стопа грешке се израчунава директно коришћењем последње једначине. Уколико се стопа грешке не повећава након одсецања чвора у стаблу одлучивања, такво одсецање се треба прихватити.

Песимистично одсецање (енг. *Pessimistic Pruning*) избегава потребу за креирањем скупа за одсецање, као и потребу за унакрсном провером након одсецања. Уместо тога, песимистично одсецање користи песимистичне статистичко узајамне тестове. Основна идеја је да стопа грешке процењене коришћењем скупа података за формирање стабла није довољно поуздана. Уместо ње, треба да се користи реалнија мера, позната као континуална корекција за биномну расподелу:

$$\varepsilon'(T,S) = \varepsilon(T,S) + \frac{|leaves(T)|}{2 \cdot |S|}$$

Оваква корекција и даље производи оптимистичну стопу грешке. Као последицу тога потребно је узети у обзир одсецање унутрашњег чвора *t* уколико његова стопа грешке потпада у једну станардну грешку стабла на које показује, наиме:

$$\varepsilon'(pruned(T,S),S) \leq \varepsilon'(T,S) + \sqrt{\frac{\varepsilon'(T,S) \cdot (1 - \varepsilon'(T,S))}{|S|}}$$

Последњи услов је заснован на статистичком интервалу поверења за пропорцију. Обично се последњи услов користи тако да *T* показује на подстабло чији је корен унутрашњи чвор *t* и *S* представља део скупа података за формирање стабла који се односи на чвор *t*. Процес песимистичног одсецања примењује *top-down* обилазак стабла. Ако је неки унутрачњи чвор одсечен, онда се сви његови потомци уклањају из процеса одсецања, резултујући веома брзим одсецањем стабла.

Одсецање засновано на грешци (енг. *Error-based pruning*) представља једну од техника одсецања и као такво је имплементирано у одређеним алгоритмима за креирња стабла одлучивања. Процена стопе грешке добија се коришћењем горње границе интервала статистичке поузданости за пропорције.

$$\varepsilon_{UB}(T,S) = \varepsilon(T,S) + Z_\alpha \cdot \sqrt{\frac{\varepsilon(T,S) \cdot (1 - \varepsilon(T,S))}{|s|}}$$

Где $\varepsilon(T,S)$ означава стопу погрешне класификације стабла T над тренинг скупом података S. Z је инверзна стандардна нормална кумулативна расподела, а α представља жељени ниво значајности. Уколико се уведе да *subtree(T,t)* означава подстабло са *root* чвором *t*, да *maxchild(T,t)* означава најфрекветније дете чвор чвора *t* и да S_t означава све инстанце у S које досежу до чвора *t*, у том случају процедура извршава *bottom-up* обилазак свих чворова и врши поређење следећих вредности:

- $\varepsilon_{UB}(subtree(T,t),S_t)$;
- $\varepsilon_{UB}(pruned(subtree(T,t),t),S_t)$;
- $\varepsilon_{UB}(subtree(T,maxchild(T,t)),S_{maxchild(T,t)})$.

На основу најниже вредности, процедура или оставља стабло онаквим какво јесте, врши одсецање чвора *t*, или врши замену чвора *t* подстаблом чији је корен у *maxchild(T,t)*.

Циљ истраживања у неколико студија је било поређење перформанси различитих техника одсецања стабла [82], [87], [88]. Резултати показују да неке методе (као што су одсецање сложености трошка, одсецање умањене грешке) теже да претерано одсечу стабло. Овакве методе као резултат одсецања креирају мање стабло одлучивања, а за које постоји могућност да је мање тачности. Друге методе (као метод одсецања заснован на грешци, одсецање песимистичне грешке и одсецање најмање грешке) теже да недовољно одсеку стабло. Закључак већине истраживања примењених на поменуте методе одсецања стабла јесте да не постоји метода одсецања која је у било ком случају боља од осталих метода одсецања.

У циљу конструисања оприналног стабла одлучивања, минимизацијом грешке генерализације могу се користити такозвани индуктор алгоритми који врше аутоматску конструкцију стабла одлучивања из датог скупа података [5]. Поред минимизације грешке генерализације, могу се дефинисати и друге већ поменуте функције попут минимизације броја чворова или минимизације просечне дубине стабла. Индуковање оптималног стабла одлучивања и датог скупа података сматра се тешким задатком. Показало се да је проналажење минималног стабла

одлучивања које је у складу са тренинг скупом података *NP*-тешки проблем [89]. Такође, конструкција минималног бинарног стабла одлучивања, у односу на очекивани број тестова потребних за класификацију нове инстанце, је *NP*-комплексни проблем [90]. Чак и проналажење минималног еквивалентног стабла одлучивања датом стаблу одлучивања или креирање оптималног стабла одлучивања из табела одлучивања може се сматрати *NP*-тешким проблемом [91]. Алгоритми који се примењују на креирање стабла одлучивања показују добре резултате када се ради о малим проблемима и мањим скуповима података. Уколико се ради о великим скуповима података, потребно је додатно применити хеуристичке методе на решавање проблема. Сви алгоритми креирања стабала одлучивања базирају се на *top-down* или *bottom-up* методу, као што је и раније сугерисано. Исто тако, два алгоритма *C4.5* и *CART* се издвајају по томе што се састоје од две концептуалне фазе: фазе раста и фазе одсецања. Остали алгоритми изводе само фазу раста. Неки од алгоритама који се најчешће користе описани су наставку овог поглавља.

***ID3* алгоритам** представља јако једноставан алгоритам за креирање стабла одлучивања. Код овог алгоритма *information gain* се користи као метод дељења. Раст стабла престаје када све инстанце припадају једној вредности циљне карактеристике или када најбољи *information gain* није већи од нуле. Тако, крај извршења долази уколико све инстанце података имају исту вредност класификације. На овакав начин доношење одлуке је једноставно, с обзиром да ће се сви тренинг подаци поклопити са одабиром групе којој припадају. Други случај у коме долази до прекида даљег креирања стабла јесте уколико су сви атрибути исцрпљени. Ово практично значи да нема више атрибута на основу ког би се вршило даље дељење [5]. *ID3* алгоритам сваки од атрибута приликом креирања стабла користи максимално једном на датој путањи стабла. Једном када се достигне максимум, уколико преостали подаци не одговарају истој класификацији, алгоритам је приморан да донесе коначну одлука. Коначна одлука се најчешће завршава тако што се дати подаци додељују најфреквентнијој класификацији, с обзиром да је даље дељење немогуће. Ни једна од поменутих ситуација не може се јавити на самом почетку рада алгоритма, јер не би било праткично да све инстанце података припадају једној класификацији.

Овај алгоритам приликом креирања стабла одлучивања не примењује методе пруновања стабла. Такође, веома тешко ради са нумеричким атрибутима и недостајућим вредностима. На основу датог скупа атрибута *ID3*, алгоритам помоћу *information gain* метода врши одабир *root* атрибута, од кога ће почети креирање стабла одлучивања. Овај алгоритам може вршити конверзију континуалних вредности атрибута у дискретне вредности. Ради што успешнијег креирања стабла одлучивања потребно је да инстанце података буду организоване као парови атрибут вредност. Такође, циљна функција има дискретну вредност излаза, док вредноси атрибута требају бити номиналне. Како би се елиминисали недостаци *ID3* алгоритма који потичу од особине овог алгоритма да врши одабир атрибута са више вредности за чвор одлуке, уведене су неке модификације овог алгоритма. У унапређеној верзији *ID3* алгоритма врши се израчунавање функције повезаности за сваки од атрибута. Добијене вредности се надаље користе у израчунавању нормализовене вредности *information gain*-а за сваки атрибут. Овако добијене нормализоване вредности се комбинују са иницијалним *information gain*-ом како би се добила нова вредност *gain*-а за сваки атрибут понаособ, што се у наставку користи као стандард за доношење одлуке. Нормализован *gain*, функција повезаности и слични методи користе се у процесу израчунавања важности сваког атрибута у скупу података. Функција повезаности не само да веома добро руководи независношћу основног *ID3* алгоритма, већ и јасно представља релацију између елемената и атрибута [92]. На овакав начин, креирањем ефективнијих правила на којима се базира креирање стабла одлучивања, повећаће се тачност *ID3* алгоритма. Уколико се упореде основна и унапређена верзија *ID3* алгоритма на основу сложености обе верзије поједини аутори закључију да је унапређена верзија временски сложенија али се временска сложеност може занемарити због доступности квалитетног хардвера на коме се овај алгоритам може изрвршавати.

***C4.5* алгоритам** представља једну од еволуција *ID3* алгоритма. Први пут је презентован у раду [47]. Овај алгоритам користи *gain ratio* као критеријум поделе. Према овом критеријуму даље дељење или гранање стабла престаје када број инстанци података које је потребно даље делити падне испод унапред дефинисаног прага. Што се тиче одсецања креираног стабла одлучивања у односу

на *ID3* алгоритам, овде је могуће применити методе одсецања. На стабла креирана помоћу *C4.5* алгоритма успешно се примењује *error-based* прунинг метода. Мотоде прунинга или одсецања стабла одлучивања се примењују након фазе раста стабла. Такође, за разлику од *ID3* алгоритма, овај алгоритам има могућности за обраду нумеричких атрибута, што у великој мери повећава опсег скупова података које је могуће обрадити. Сем тога, *C4.5* алгоритам успешно врши обраду скупова података са недостајућим вредностима атрибута, што значи да се може индуковати из тренинг скупа података који садржи недостајуће вредности помоћу коригованих *gain* критеријума. *C4.5* алгоритам обезбеђује скалабилност. У поређењу са *ID3* алгоритмом разликује се у начину рачунања информационе добити. Номинални атрибути имају онолики број деце колико има различитих вредности циљног атрибута, док се код нумеричких атрибута услову поделе додељују вредности. Да би се вршила подела по нумеричком атрибуту потребно је дискретизовати га, односно одредити граничну вредност за функцију поделе. Вредност за поделу се бира тако да атрибут има највећу информациону добит. Елементи се деле у два скупа у односу на то да ли испуњавају услов поделе или не. Информациона добит за атрибуте се рачуна тако што се сортира скуп елемената по вредности атрибута за који се рачуна информациона добит. Након тога се пролази кроз скуп података и рачуна се информациона добит, под претпоставком да тренутни елемент дели скуп података [93]. Поступак се понавља за сваки елемент скупа и чува се позиција и вредност атрибута оног елемента код кога је постигнута највећа вредност информационе добити.

***J48* алгоритам** је екстензија *ID3* алгоритма за креирање стабла одлучивања. Тачније, *J48* класификатор се може посматрати као *java* имплементација *C4.5* алгоритма. Додатна функционлност која га разликује од *ID3* алгоритма је обрада недостајућих вредности, могућност одсецања креираног стабла одлучивања, рад са атрибутима који имају континуалан опсег вредности, креирање правила одлучивања на основу путања у стаблу одлучивања итд. *J48* алгоритам је првенствено имплементиран у склопу *WEKA* data mining алата, о коме ће бити речи у наредним поглављима. У случају креирања превеликог стабла одлучивања, за разлику од других алгоритама, J48 нуди могућност одсецања са циљем повећања прецизности [94], тако да додатна провера прецизности након одсецања

није потребна. Код других алгоритама се класификација изводи рекурзивно све до момента док сваки појединачни лист није чист. Класификација података треба да буде што је могуће савршенија. Овај алгоритам генерише правила на основу којих се сваки појединачни идентитет података генерише. Циљ алгоритма је прогресивна генерализација стабла одлучивања све до момента док стабло одлучивања не достигне равнотежу између флексибилности и тачности. Основни кораци *J48* алгоритма могу се дефинисати на следећи начин:

- Уколико инстанце података припадају истој класи, у том случају целокупно стабло је заправо лист, те се тај лист враћа тако да указује на саму класу;
- Потенцијална информација се израчунава за сваки арибут, тестирањем истог. Након тога се на основу резултата тестирања рачуна информациона добит;
- Након тога, на основу критеријума селекције, проналази се најбољи атрибут на основу кога ће се вршити гранање;

J48 алгоритам подједнако добро ради са континуалним и дискретним атрибутима. Вредност прага код континуалних атрибута се одређује исто као код *C4.5* алгоритма. На овакав начин врши се подела скупа података на оне чији атрибути имају вредност испод унапред дефинисаног прага и оне чији атрибути имају вредност изнад или једнаку унапред дефинисаном прагу. Приликом одсецања делова стабла применом овог алгоритма врши се одсецање оних делова стабла који не досежу до чворова који представљају листове.

***CART* алгоритам (енг. *Classification and Regression Trees*)** се карактерише особином да креира бинарна стабла одлучивања у којима сваки унутрашњи чвор има тачно два излазна потега [85]. Када је доступно, *CART* може да узме у обзир трошак погрешне поделе на индукованом стаблу. Битна предност *CART* стабла је његова могућност да генерише регресивна стабла. Регресивна стабла су стабла код којих чворови предвиђају тачан број, не класу. У случају регресије, *CART* тражи поделе које минимизују предвиђање квадратне грешке, тачније најмање квадратно одступање. Учесталост вредности циљног атрибута на скупу елемената рачуна се исто као и у друга два алгоритма.

Дефинисана функција користи претходно утврђен број атрибута за које се рачуна однос добити. Уколико је број атрибута већи од унапред дефинисаног броја бира се случајни подскуп и само на том подскупу се бира најбољи атрибут, тачније онај који има највећи однос добити. Ако број атрибута није већи од утврђеног броја рачуна се однос добити за све атрибуте и бира се најбољи. Стабло које се гради је бинарно. Након тога се проверава да ли постоји атрибут са највећим *gini* односом и чува атрибут који има највећи *gini* однос, као и вредност поделе која се бира тако да атрибут има највећи *gini ratio*. Када се ради са номиналним атрибутима бира се вредност тако да атрибут има највећи *gini ratio*. Скуп елемената се дели по услову да ли је вредност атрибута једнака вредности по којој се атрибут дели или не. Код нумеричких атрибута се бира и вредност тако да атрибут има највећу вредност за *gini ratio*. Скуп елемената се дели по томе да ли је вредност атрибута мања од вредности поделе или не. Приликом поделе скупа елемената за сваки чвор дете бира се најчешћа циљна вредност атрибута на подскупу елемената, затим се проверава да ли постоји чвор који би га заменио и, уколико постоји, смешта се чвор у ред за поделу. Наставља се све док има чворова учења или док стабло не достигне довољан број чворова.

Процена тачности класификатора

Сам квалитет добијеног класификатора се потврђује резултатима насталим тестирањем класификатора на неозначеним инстанцама (инстанце којима су познате вредности предикторских атрибута, али је непозната вредност ознаке класе).

Евалуација класификатора се најчешће темељи на предвиђању тачности (проценат тачних предвиђања подељен са укупним бројем предвиђања). Уобичајене технике на основу којих се рачуна класификациона тачност су:

- *Holdout* (у коме се врши подела расположивог скупа за тренинг у одређеним односима на скуп за тренинг, тестирање и валидацију);
- подела скупа у одређеним процентима (енг. *percentage split*):
- метода унакрсне валидације (дели скуп на *N* делова (fold)).

Проблем настаје зато што у пракси често постоји само један скуп података одређене величине и све процене морају бити добијене на основу тог скупа што доводи до колизије, с обзиром на то да је истовремено потребан што већи скуп за обучавање (да би се добио добар класификатор) и адекватан скуп за тестирање (за добру процену о грешци), а све на основу једног почетног скупа података. Из наведеног произилази да је довољно наглашен значај обима и квалитета улазних података у контексту резултата data mining-a.

Holdout метод дели скуп података на два међусобно независна подскупа, тренинг подскуп који се користи за обуку модела и тест подскуп података. Скуп података се дели тако да се 2/3 инстанци полазног скупа података користи за обуку класификационог модела, а 1/3 за оцену тачности предвиђања. Процена тачности предвиђања је случајан број који зависи од поделе инстанци на тренинг и тест скуп. Поступак се понавља *k* пута и тачност се може добити као средња вредност. Стандардна девијација се добија као стандардна девијација појединачних оцена. Тачност предвиђања наученог скупа правила повећава се са повећавањем броја примерака у тренинг скупу. Издвајањем већег броја инстанци података за тестирање, повећава се пристрасност процене, док се смањивањем скупа инстанци проширују границе интервала поверења.

Метод унакрсне валидације (енг. *Cross-validation*) дели скуп података на *к* међусобно искључиих скупова података приближно исте величине. Поступак учења и оцењивања се понавља *к* пута, где се сваки пут користи један подскуп као тест скуп података. Процена тачности предикције унакрсним оцењивањем је случајан број који зависи од поделе инстанци на подскупове. На крају процене од *к* добијених резултата узима се средња вредност као једна јединствена процена. Предност оваквог метода над сличним методима је томе што се све инстанце података користе као у тренинг процесу тако и у процесу валидације. Поред тога, свака од инстанци података се користи у процесу валидације само једном. Како *к* не представља фиксни параметар, одабиром истог може се делити скуп података на више или мање делова. У пракси се најчешће користи *к=5* и *к=10*. Унаксна валидација, као метод, налази примену код поређења перформанси различитих предикционих процедура примењих на креирање предиционих модела за

истоветни скуп података. Примера ради, ова метода оцењивања се може користити за поређење перформанси метода *к*-најближих суседа и *SVM* метода.

Основни појам при процени тачности класификационих модела је појам грешке. Грешком се сматра погрешна класификација, што значи да примена класификационог модела на одабрану инстанцу података резултира предикцијом припадности класи која је различита од стварне класе дате инстанце. Уколико се свакој грешци придаје једнак значај, тада је укупан број грешака на посматраном скупу примерака добар индикатор успеха или неуспеха класификационог модела. Тачност предикције је основни показатељ перформанси креираног класификационог модела који представља проценат правилно класификованих нових примерака коришћењем научених правила. Два основна недостатка мере тачности односе се на занемаривање ризика између типова грешака и зависности од дистрибуције класа у скупу података, а не од карактеристика инстанци. Тачније, у већини практичних ситуација је врло важно разликовати одређене типове грешака [95].

У случајевима када је потребно разликовати више типова грешака резултат класификације може се приказати у облику дводимензионалне матрице грешака. У овако креираној матрици сваки ред матрице одговара једној класи и садржи број инстанци којима је предикцијом одређена дата класа. Свака колона матрице је означена са једном класом и садржи број примерака којима је то стварна класа. Број тачно класификованих инстанци се налази у дијагонали матрице. Сви остали елементи матрице означавају број инстанци које су неиправно класификоване као нека од преосталих класа. Највећи број мера за процену класификационих модела односи се на класификационе проблеме са две класе. Ово не представља ограничење за употребљивост самих мера процене, јер се проблеми са већим бројем класа могу приказати у облику низа проблема са две класе. На овакав начин се из сваке групе од по две класе врши издвајање једне класе као циљне, а скуп података се дели на позитивне и негативне примерке циљне класе. Негативне примерке циљне класе чине примерци свих осталих класа. Постоје четири могућа резултата предвађања:

- стварно позитивни (енг. *True Positive Rate - TP*) исходи;

- стварно негативни (енг. *True Negative Rate - TN*) исходи;
- лажно позитивни (енг. *False Positive Rate - FP*) исходи;
- лажно негативни (енг. *False Negative Rate - FN*) исходи.

Прва два могућа резултата су исходи који представљају исправну класификацију, док друга два исхода представљају два могућа типа грешке приликом класификације. На основу дефинисаних мера издвајају се одзив и специфичност као две додатне мере. Одзив и специфичност спадају у мере за процену класификационих модела које разликују два споменута типа грешке. Одзив као мера позната је и под називом сензитивност. Ове две мере заснивају се на истом принципу, као и тачност с тим што сензитивност мери тачност у позитивним примерцима, а специфичност представља тачност у негативним примерцима.

$$Odziv = \frac{TP}{TP + FN}$$

$$Specifičnost = \frac{TN}{TN + FP}$$

Још један пар мера који је у употреби приликом процене креираног предикционог модела чине тачност и прецизност. Овај пар мера осетљив је на различите типове грешака. Најчешће се користи у проблемима код којих је број стварно позитивних доста мањи у односу на број стварно негативних примерака. Тачност и прецизност могу се дефинисати на следећи начин:

$$Tačnost = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Preciznost = \frac{TP}{TP + FP}$$

Одзив је дефинисан као тачност у позитивним примерцима, док се прецизност дефинише као тачност у позитивној прогнози циљне класе. У контексту проблема проналажења информација, одзив представља однос пронађених релевантних докумената у односу на укупан број релевантних докумената, док је прецизност удео релевантних докумената у укупном броју пронађених докумената.

Позитивна и негативна предиктивна вредност су још један пар мера који уважава различите типове класификацијских грешака. Заснивају се на принципима прецизности, тј. позитивна предиктивна вредност представља прецизност у позитивним примерцима, док је негативна предиктивна вредност прецизност у негативним примерцима [95].

Осим наведених мера, постоје и мере које се не заснивају на фиксирању једне компоненте мера. У овакве мере спада и *F-mera* која се дефинше као:

$$F - mera = \frac{2 \cdot Odziv \cdot Preciznost}{Odziv \cdot Preciznost}$$

Квалитет класификацијског модела изражен једним бројем или паром бројева не може у потпуности илустровати варијабилност решења која се могу извести употребом одабране технике моделирања. Међусобне зависности различитих мера и/или параметара боље се могу описати употребом графова којима је могуће приказати велики распон различитих могућности. Једна од графички оријентисаних мера класификацијског модела је *ROC* (енг. *Receiver Operating Characteristic*) анализа, преузета из области детекције сигнала [96]. За разумевање ROC анализе најважнији је појам граничних вредности поузданости одређеног класификатора. Сензитивност и специфичност су две мере које карактеришу специфичну тачност класификатора у позитивним и негативним примерцима. Када је гранична вредност поузданости постављена високо, тј. наглашен критеријум поузданости класификатора, сензитивност класификатора ће бити ниска, а специфичност висока. Ако се критеријум поузданости снизи, сензитивност ће порасти, а специфичност опадати. На тај начин, два класификациона модела могу се упоредити преко широког спектра поузданости генеришући криву која описује зависност броја стварно позитивних примерака од броја лажно позитивних примерака детектованих моделом. Уколико се са *T = (x, y)* означи тачка на ROC кривој посматраног класификатора, вредности координата *x* и *y* израчунавају се помоћу израза:

$$x = \frac{FP}{FP + TN}$$

$$y = \frac{TP}{TP + FN}$$

Дакле, вертикална оса *ROC* криве означава број стварно позитивних примерака израженим као удео у укупном броју позитивних примерака. С друге стране, хоризонтална оса *ROC* криве означава одговарајући број лажно позитивних примерака изражен као удео у укупном броју негативних примерака. Облик *ROC* криве неког класификатора зависи од скупа примерака који је анализиран.

4.4.3 Кластеризационе data mining технике

Фундаментална претпоставка од које се полази када се посматра примена техника кластеризације у претраживању информација и обради великих скупова података је хипотеза кластеризације која гласи: „сви скупови података у једном кластеру имају сличне особине на основу којих припадају датом кластеру, али и релевантност података који се налазе у скупу података мора бити иста“ [97]. Оваква претпоставка значи да уколико је један скуп података из кластера релевантан на задати упит онда је вероватно да су и други скупови података из истог кластера такође релевантни за исти упит. Ово произилази из чињенице да се у једном кластеру налазе скупови података који деле многе термине. Још једна хипотеза од које се полази је: „везе између скупова података преносе информацију о релевантности скупа података на задати упит“ [98]. Уколико је ово тачно, подразумева се да ће кластеризација довести до побољшања претраживања информација на корпусу докумената. Ово доводи до закључка да ће документи који су слични међусобом бити у истом кластеру, уколико су они правилно кластеризовани. Из ових закључака и дефиниција се види да је суштина иста и да постоје основани разлози за увођење кластеризације у процес претраживања информација.

На основу параметара које користе, начина обраде скупа података и начина организације података у оквиру кластера, алгоритми кластеризације се могу грубо поделити на нивовске алгоритме и хијерархијске алгоритме кластеризације. Нивовска кластеризација креира нивое кластера без неке посебне структуре која би ове нивое кластера повезала међусобно. Као излаз хијерахијске кластеризације

добија се структура која носи много више информација од неструктуиране групе кластера која се добија као резултат нивовске кластеризације.

4.4.3.1 K-means алгоритам кластеризације

K-means алгоритам кластеризације је један од најзначајнијих нивовских алгоритама кластеризације. Циљ овог алгоритма је да минимизује просечно квадратно Еуклидово растојање података од центра кластера коме дати подаци припадају. Минимизација просечног квадратног Еуклидовог растојања може се описати следећом једначином:

$$\vec{\mu}(\omega) = \frac{1}{|w|}\sum_{\vec{x}\in\omega}\vec{x}$$

У датој једначини центар кластера је означен са $\vec{\mu}$, док је кластер који садржи податке означен са ω. Дефиниција *k-means* алгоритма кластеризације претпоставља да су подаци представљени као вектори нормализованих дужина у реалном простору на сличан начин. Идеалан *k-means* кластер је сфера са центром кластеризације у центру гравитације. Идеално је и ако се кластери не преклапају. Мера која показује колико добро елементи, одабрани за центар кластера или такозвану центроиду, представљају чланове кластера је резидуална сума квадрата или краће RSS. То је сума квадрата растојања свих вектора од центоиде и може се описати следећим двема једначинама.

$$RSS_k = \sum_{\vec{x}\in\omega_k}|\vec{x} - \vec{\mu}(\omega_k)|^2$$

$$RSS = \sum_{k=1}^{K} RSS_k$$

RSS је објективна функција у *к-means* алгоритму и тежи се ка томе да се она минимизује. Како је *N* фиксно, минимизација *RSS*-а је еквивалентна минимизацији средњег квадратног растојања, односно мере колико добро центроиде представљају своје податке. *N* представља количину података у кластеру. Рад алгоритма се своди на два корака. У првом кораку алгоритам селектује иницијалне центре кластера *K* рандом селектованих скупова података.

У наставку алгоритам помера центре кластера по простору у циљу минимизовања *RSS*-а. Након тога се ради итеративно понављање по два корака док се не достигне услов за крај итерације, а то је понављање скупа података за кластер са најближом центроидом. Када се достигне услов за крај итерације ради се прерачунавање центроида на основу тренутних чланова у кластеру. Овде се примењују различити услови прекида.

Иако *к-means* алгоритам кластеризације има велике предности у погледу лаке имплементације, одликују га два велика недостатка [99]. Први од недостатака је брзина извршења, с обзиром на то да се у сваком кораку мора израчунати растојање између сваке тачке и сваког кластера, што за велике скупове података може представљати велики утрошак времена. Други недостатак представља одабир почетних кластера, као и почетних центара кластера, с обзиром на то да су перформансе *k-means* алгоритма зависне од самог почетног одабира.

4.4.3.2 K-medoids алгоритам кластеризације

Ефикасност *k-means* алгоритма кластеризације може бити мања уколико скуп података садржи *outlier*-е. Присутност екстремних или погрешних вредности у скупу податка које заправо и чине *outlier*-е утиче на израчнавање вредности центроида које се у највећем броју случајева представљају просечну вредност. Из тог разлога се веома често користи *k-medoids* алгоритам кластеризације који је знато робуснији и отпорнији на *outlier*-е и шум у подацима. Овај алгоритам уместо коришћења просечне вредности за центориде кластера у почетној фази врши одабир једне од тачака у скупу података и њу проглашава за центар кластера. На овакав начин такозвана *medoid*-а је објекат у кластеру података који је најближи центру и притом има минималну суму растојања од осталих тачака у кластеру [100]. Као што је случај са *k-means* алгоритмом, циљ *k-medoids* алгоритма је, такође, минимизација квадратне грешке растојања. Основна идеја овог алгоритма је да најпре израчуна *k* репрезентативних *medoid*-а. Након тога, сваки објекат скупа података додељује се најближој *medoid*-и, штозначи да ће објекат i бити додељен кластеру w_i, уколико је *medoid*-а mw_i најближа овом објекту у односу на остале *medoid*-е m_w.

Репрезентативни алгоритми који припадају овој групи кластеризационих алгоритама су *Partitioning Around Medoids (PAM), Clara, Clarans* итд. Основна идеја PAM алгоритма је селекција *к* репрезентативних тачака око којих ће се формирати иницијални кластери. Након одабира иницијалних кластера врши се померање тачака одабраних за центре кластера како би се оформила боља репрезентација кластера. Све могуће комбинације репрезентативнинх и нерепрезентативних тачака се анализирају и квалитет резултујућих кластера се израчунава за сваки пар тачака. Иницијално одабрана тачка која је репрезент целог кластера се мења новом тачком, што доводи до највећег смањења функције дисторзије. Процес замене нерепрезентативне тачке репрезентативном тачком врши се тако што се за сваку од нерепрезентативних тачака проверавају четири могућа случаја [101]. Уколико се нерепрезентативна тачка означи са p_{rand}, репрезентативна са p_i, поменута четири случаја за конкретну нерепрезентативну тачку *p* могу се дефинисати као:

- Први случај: p оригинално припада репрезентативној тачки p_i. Ако је након замене p најближа једној од осталих репрезентативних тачака p_j тада се *p* додељује p_j;
- Други случај: p оригинално припада репрезентативној тачки p_i. Ако је након замене *p* најближа p_{rand} онда се *p* додељује p_{rand};
- Трећи случај: p оригинално припада једној од осталих репрезентативних тачака p_j. Ако је након замене p још увек најближа једној од осталих тачака p_j, онда нема потребе за додељивањем нове вредности *p;*
- Четврти случај: *p* оригинално припада једној од осталих репрезентативних тачака p_j. Ако је након замене *p* најближа p_{rand}, онда се p додељује p_{rand}.

Cost функција је дефинисана као промена вредности функције дисторзије приликом замене репрезентативне тачке нерепрезентативном тачком. Укупна вредност *cost* функције замене *C* представља збир појединачних *cost* вредности свих нерепрезентативних тачака. Уколико је вредност cost функције негативна, замена вредности тачака је дозвољена, с обзиром да негативна вредност означава редуковање функције дисторзије. *PAM* алгоритам кластеризације је ефикасан за

мале скупове података, али није скалабилан када се ради са великим скуповима података. Временска сложеност овог алгоритма је $O(K(N-K)^2 I)$.

Два алгоритма кластеризације издвајају се као унапређење *PAM* алгоритма када је потребно скалирање како би се радило са великим скупом података. Један од њих је кластерзација великих апликација (*CLARA*) и кластеризација великих апликација на основу рандом претраге (*CLARANS*). *CLARA* се базира на једноставној стратегији: рандом одабиром се издваја мали део полазног скупа података и над издвојеним делом изводи се PAM алоритам како би се из креираног узорка пронашао довољан број *K medoid*-а. Уколико издвојени узорак може приближно представити оригинални скуп података, пронађене репрезентативне *medoid*-е биће добра апроксимација *medoid*-а које би се пронашле претрагом целог скупа података. Како би се унапредио квалитет кластеризације *CLARA* алгоритам се употребљава над већим бројем рандом креираних подскупова података. Уколико претпоставимо да је величина подскупа *M*, и да је *M* мање од величине оригиналног скупа података означене са *N*, у том случају сложеност *CLARA* алгоритма је $O(\big(KM^2 + K(N-K)\big)I)$. Посматрано са стране перформанси *CLARA* је ефикаснији алгоритам кластеризације од *PAM* алгоритма. Међутим, тачност кластеризације и креирање најбољег кластера зависни су од пронађених *medoid*-а. Уколико било која од *medoid*-а пронађена из подскупова података није међу *K* најбољих *medoida*, креирана кластеризација засигурно није најбоља могућа. CLARANS као други алгоритам кластеризације унапређује квалитет CLARA алгоритма кластеризације. Овај алгоритам се може моделирати као претрага стабла где чвор представља скуп *K-medoid*-а, а суседни чворови се разликују за једну *medoid*-у, тако да сваки чвор има $K(N-K)$ суседа. Евалуација сваког од чворова се врши применом функције дисторзије како би се измерио квалитет кластера. *CLARANS* на почетку врши рандом селектовање чвора и његових суседа. Уколико суседни чвор показује бољи квалитет кластеризације, прелази се на дати чвор и наставља се итеративни поступак. Уколико дати чвор представља локални минимум и уколико ни један тестирани сусед не даје бољу кластеризацију процедура се покреће од почетка са новим рандом селектованим чворовима. Као критеријум за крај итерације обично се узима унапред дефинисан број локалних минимума.

4.4.3.3 Хијерархијска кластеризација

Подаци у хијерархијској кластеризацији нису унапред подељени у одређени број кластера. Наиме, кластеризација се састоји од низа партиција које могу започети са једним кластером који садржи све податке, све до *n* кластера који се састоје од само једног податка. Технике хијерархијске кластеризације се могу поделити на агломеративне методе и методе дељења. Алгомеративне методе врше спајање *n* појединачних података у групе, док методе дељења врше дељење скупа од *n* података у мање групе.

Алгоритми хијерархијског агломеративног кластеровања (*HAC*) проналазе кластере иницијалним додељивањем сваког објекта одговарајућем кластеру, затим се врши стално мешање парова кластера док се не достигне одговарајући критеријум прекида алгоритма. За одређивање парова кластера који ће учествовати у мешању користе се различите функције које раде на неким од критеријума кластеризације.

Ови алгоритми кластеризације могу бити *bottom-up* и *top-down*. *Bottom-up* алгоритми на почетку третирају сваки скуп података као јединствен кластер, затим сукцесивно мешају парове кластера све док се сви кластери не нађу у једном кластеру и на тај начин садрже све податке. Назив агломеративна кластеризација потиче баш од *bottom-up* алгоритама хијерархијске кластеризације. *Top-down* кластеризација захтева метод цепања кластера. Цепање кластера се обавља рекурзивно док се не дође до индивидуалног податка. *HAC* се много више користи у претраживању информација него *top-down* кластеризација. Код кластеризације *n* докумената, након *l* корака спајања, решење садржи тачно *n-l* кластера, а сваки корак мешања смањује број кластера који се добијају на излазу за један. Парови кластера који су одабрани за наредну фазу мешања воде *(n–l-1)* решењу кластеризације које оптимизује одређену функцију критеријума кластеризације. На овај начин се сваки од могућих $(n - l)\cdot(n - l - 1)/2$ парова који учествују у мешању одређује, а један који води решењу кластеризације са максималном или минималном вредношћу функције критеријума се селектује.

Алгоритам се заснива на два посебна корака. Оба корака су велике сложености. Први корак је израчунавање сличности између свих парова података у скупу

података. Сложеност овог корака износи $O(n^2m)$, где је n број података, а m број термина. Други корак је поновна селекција пара кластера који најбоље оптимизују функцију критеријума. Најочигледнији начин да се ово уради је да се одреде предности добијене спајањем сваког пара кластера након сваког нивоа агломерације и да се одабере пар који највише обећава. Извођење l-тог корака агломерације захтева време израчунавања од $O((n-l)^2)$, што доводи до опште сложености од $O(n^3)$. Сложеност овог корака може се смањити, јер се побољшања условљена функцијом критеријума добијена мешањем кластера i и j не мењају током различитих корака агломерације све док i и j нису одабрани за процес мешања. Из тога следи да се различите повољне вредности функције могу израчунати само једном за сваки пар кластера и могу се сместити у ред приоритета. Ако се пар кластера i и j одабере за мешање како би се добио кластер p, тада се ред заснован на приоритету обнавља, тако што се свако побољшање које одговара кластеру i или кластеру j уклања из овог реда, а побољшања добијена мешањем остатка класера са новоформираним кластером p се уписују у ред. Хијерархијска кластеризација не захтева фиксан предефинисан број кластера. Међутим, у неким апликацијма је потребно извршити поделу дисјунктних кластера, као што је то случај у нивовским кластерима. У тим случајевима процес хијерахијске класеризације мора бити прекинут у одређеном тренутку. Прекидна тачка у алгоритму кластеризације дефинисана је критеријумом прекида [102].

Методе дељења, као и агломеративне методе имају недостатак, у смислу да након што је направљена партиција скупа у кластере, не постоји могућност премештања јединица из једног кластера у други кластер. Међутим, ако су од интереса већи кластери, тада методе дељења имају предност над агломеративним методама, у којима се већи кластери достижу само након великог броја корака. Генерално посматрано, могу се издвојити два скупа алгоритама дељења: *monothetic* и *polythetic*. У *monothetic* проступу подела групе на две подгрупе је заснована на једној променљивој, док се у *polythetic* приступу користи p варијабли да би се направило раздвајање. Ако су променљиве бинарне, *monothetic* приступ се може једноставно применити. Код *monothetic* приступа подела на две групе се заснива на присуству или одсуству атрибута. Овакав начин тежи минимизацији броја партиција које тек требају бити направљење. Један пример критеријума

хомогености је информацијски показатељ, *C*, дефинисан са p променљивих и n објеката.

$$C = pn \log n - \sum_{k=1}^{p} [f_k \log f_k - (n - f_k) \log(n - f_k)],$$

где је f_k број појединаца који садрже *к*-ти атрибут. Ако групу X треба раздвојити у две групе A и B, тада је редукција у C једнака $C_x - C_A - C_B$. Идеалан скуп кластера требло би да садржи чланове са идентичним атрибутима и C једнаким нули. Уместо хомогености кластера, атрибут коришћен у сваком кораку се може изабрати с обзиром на његову целокупну повезаност са свим осталим атрибутима у том кораку. Сваким се кораком партиција скупа одваја, с обзиром на присутност или одсутност атрибута чија је повезаност са осталима максимална. Атрибут се бира тако да максимизира x^2 вредност или неки информациони показатељ. Једна од највећих предности *monothetic* методе дељења је познавање који атрибут и у којој фази ће довести до раздвајања кластера. Проблем оваквих метода је да поседовање одређеног атрибута који је редак или се тешко проналази у комбинацији са осталима атрибутима, а на коме се базира дељење, може довести до лошег финалног решења.

Polythetic методе дељења су сличније агломеративним методама јер користе све варијанте истовремено, па самим тим могу радити на матрици удаљености. Како би се скуп података поделио применом *polythetic* метода дељења, ради се издвојеном групом података и остатком. Тражи се члан у остатку чија је просечна удаљеност од осталих чланова у остатку, умањена за његову удаљеност од издвојене групе, највећа. Ако је највећа удаљеност негативна, поступак се зауставља и самим ти подела је потпуна. Издвојена група се може креирати са чланом који има највећу просечну удаљеност од осталих чланова у групи.

***DBSCAN* (енг. *Density-based spatial clustering of applications with noise*)** алгоритам је један од алгоритама хијерахијске кластеризације у коме се број кластера за разлику од *k-means* алгоритма не одређује на почетку већ зависи од самих података које је потребно кластеризовати [103]. С обзиром да се ради о просторном кластеризационом алгоритму базираном на густини података, неке од

тачака података могу остати некластеризоване. *DBSCAN* алгоритам користи два фактора ε и *MinPts*, где ε представља максимални радијус између посматране тачке и суседних тачака, а MinPts представља најмањи број тачака података које треба да буду садржане у ε околини. Ова два фактора могу се одредити на основу самог скупа података тако да воде најбољој кластеризацији, применом адаптивног *DBSCAN* метода. Такође, уколико је скуп података добро познат, поменути фактори се могу израчунати и ручно. Принцип рада *DBSCAN* алгоритма је такав да, уколико имамо скуп тачака података у неком простору које је потребно кластеризовати, све тачке простора се могу класификовати у централне (енг. *core*) тачке, граничне (енг. *border*) и *outlier*-е [104].

- Централна тачка А представља ону тачку за коју постоји одређен број тачака података већи од MinPts, а које се налазе у ε околини тачке А;
- Гранична тачка А представља ону тачку која је директно доступна од централне тачке и притом број тачака у ε околини тачке није већи од унапред дефинисаног фактора MinPts;
- Outlier тачка је она тачка која према дефиницији није нити *core* тачка нити *border* тачка.

Доступност није симетрична релација, јер по дефиницији није могуће доћи до неке тачке од тачке која није централна тачка, без обзира на раздаљину између ових двеју тачака. Из тог разлога се уводи додатни појам повезаности гуситине како би се формално дефинисао обим кластера које је пронашао *DBSCAN* алгоритам. Према овом појму две тачке p и q су повезане по густини, уколико постоји тачка o таква да су p и q обе доступне од тачке o. Повезаност густине је према датој дефиницији симетрична релација. На овако дефинисан начин креирани кластер задовољава две особине:

- све тачке унутар кластера су узајамно повезане по густини;
- уколико је тачка доступна по густини од било које друге тачке у кластеру, онда се и она може сматрати делом кластера.

На основу свега наведеног DBSCAN алгоритам може се описати следећим корацима:

- пронаћи све тачке у ε околини сваке од тачака, и идентификовати централне тачке које имају више од MinPts суседа;
- Пронаћи све повезане компоненте централних тачака на графу суседа, не узимајући у обзир све нецентралне тачке;
- доделити сваку нецентралну тачку најближем кластеру, уколико је кластер у ε околини, у супротом тачку прогласити за *outlier*.

DBSCAN обилази све тачке унутар скупа података (нпр као кандидате за различите кластере). Из практичних разлога, временска комплексност највише зависи од позива *regionQuery*. *DBSCAN* извршава тачно један такав упит за сваку тачку, а у случају када се користи индексна структура која даје сложеност O(logn) за проналазак суседа, просечна комплексност је $O(n \log n)$ (ако се параметар ε одабере на одговарајући начин тако да се у просеку увек селектује $O(\log n)$ тачака). Без коришћења индексне структуре, или пак на дегенерисаним подацима (на пример где су све тачке на растојању које је мање од ε), најгори случај има комплексност $O(n^2)$. Матрица дистанци величине $(n^2 - n)/2$ се може прилагодити тако да се избегну поновна израчунавања растојања, али ово захтева $O(n^2)$ меморије, док имплементације које се не базирају на матрицама захтевају O(n) меморије [105].

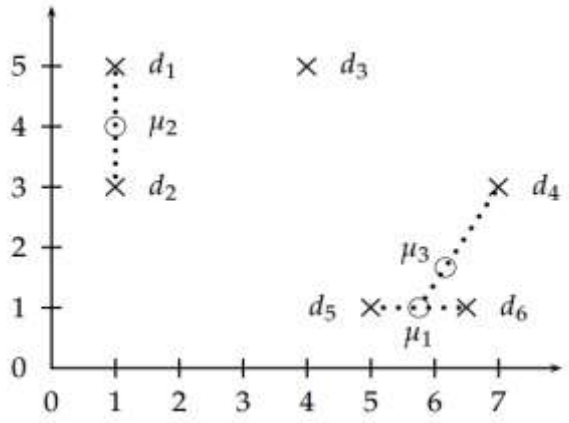
4.4.3.4 Центроидална кластеризација

Код центроидалне кластеризације сличност два кластера се дефинише на основу сличности њихових центроида. За одређивање сличности центроида може се користити једначина дата испод. Где N_i и N_j представљају број података у кластеру w_i и w_j респективно, а $SIM - CENT(w_i, w_j)$ вредност сличности. Из ове једначине се уочава да се у поступку центроидалне кластеризације искључују парови елемената који се налазе у истом кластеру.

$$SIM - CENT(w_i, w_j) = \vec{\mu}(w_i) \cdot \vec{\mu}(w_j) = \left(\frac{1}{N_i} \sum_{d_n \epsilon w_i} \overrightarrow{d_m}\right) \cdot \left(\frac{1}{N_j} \sum_{d_n \epsilon w_j} d_n\right)$$

Поступак овог метода кластеризације приказан је на слици испод. У прве две итерације алгоритма формирају се кластери $\{d_5, d_6\}$ са центроидом μ_1, и $\{d_1, d_2\}$ са

центроидом μ_2. Ова два кластера се креирају из разлога јер парови < d_5, d_6> и < d_1, d_2> имају центроиде са највећим степеном сличности.



Слика 3: Пример центроидалне кластеризације

У следећој итерацији највећа сличност центроида је између центориде μ_1 и центроиде елемента d_4, што као резултат даје кластер $\{d_4, d_5, d_6\}$ са центроидом μ_3. Центроидална кластеризација не даје као продукт најбоље мешање елемената на почетку алгоритма, као ни најбоље мешање кластера у наставку алгоритма, тачније у наредним итерацијама.

4.4.3.5 Оцена валидности креираних кластера

У општем смислу, алгоритми кластеровања дефинишу поделу скупа података засновану на одређеним претпоставкама, при чему ово не мора бити безусловно најбоља подела која уклапа скуп података. Како се на основу алгоритама кластеровања издвајају кластери који нису познати на почетку процеса коначна подела скупа података у већини апликација захтева неки вид процене. На основу једне од дефиниција, под валидацијом кластеровања се подразумева поступак оцењивања колико се добро подела датог скупа слаже са основном структуром података [106]. Финални кластери захтевају поступак процене који укључује решавање низа проблема, а који се могу дефинисати и на следећи начин:

- одређивање оптималног броја кластера;
- испитивање квалитета кластера;
- процена о томе да ли се резултујућа подела добро слаже са основном структуром података.

Из угла примене техника кластеризације над реалним скупом података, најважнија одлука у примени кластер анализе је избор одговарајуће методе кластеровања и одређивање оптималног броја кластера, јер успех даље анализе итекако зависи од ове одлуке. Сама евалуација крајњег продукта кластеризације у основи представља тежак задатак, с обзиром да не постоји универзална дефиниција добре кластеризације. Све мере валидноси се могу генерално поделити у три категорије:

- мере екстерне валидности;
- мере интерне валидности;
- мере релативне валидности.

Мере екстерне валидности се користе за процену степена слагања између две поделе (*U и V*), где је подела *U* резултат поступка кластеровања, а подела *V* је формирана на основу *a-priori* информације, независно од партиције *U* (као што је класификација). У ову групу мера спадају тачност, прецизност, одзив и ентропија [107]. Главни недостатак екстерних мера је да се не могу увек примењивати, јер у реалном скупу података a-priori информације нису увек познате. Под екстерним мерама валидности за мерење ефикасности предложене методе кластеровања, може се користити тачност (енг. *accuracy*), која је дефинисанана следећи начин:

$$r = \frac{1}{m}\sum_{i=1}^{k} max_i(m_l^i),$$

где је са m означен број елемената у скупа, са k број кластера, а са m_l^i број објеката из *i*-те класе, који припадају *l-том кластеру*. Грешка кластеровања e је дефинисана као $e = 1 - r$.

Друга од мера валидности из групе екстерних мера је ентропија [108]. Ентропија се може дефинисати као: нека је $C = \{A^1, \dots, A^k\}$ скуп дисјунктних кластера за посматрани скуп *A*, то јест важи $A = \bigcup_{j=1}^{k} A^j$. Циљ је минимизовати такозване укупне ентропије за скуп кластера, то јест очекивање ентропије дефинисано као:

$$H(C) = \sum_{i=1}^{k} \frac{m_i}{m} H(A^i),$$

где је $|A^i| = m_i$ кардиналност *i*-тог кластера, *m* је обим скупа *A*, а $H(A^i)$ је ентропија кластера A^i. Уколико је број кластера *k=1*, добија се *H(C)=H(A)*, док за *k=m* важи *H(C)=0*.

Мере интерне валидности користе информације добијене унутар поступка кластер анализе и не захтевају додатне информације о подацима. Интерни критеријуми мере хомогеност унутар кластера, раздвојеност између кластера или њихову комбинацију и представљају слагање, то јест фитовање (eng. *goodness-of-fit*) улазних података и резултата груписања података путем кластер анализе [109]. Данас у литератури постоје бројни радови који се баве анализирањем различитих мера интерне валидности кластера, као и модификацијом постојећих мера. Неке од најкоришћенијих су *Calinski-Harabasz, Gamma indeks, Silhouette indeks* и *Bajesov* информациони критеријум. *Colinski-Harabasz* критеријум се може израчунати као [110]:

$$CH(k) = \frac{tr(B)/(k-1)}{tr(W)/(m-k)},$$

где је *к* број кластера, *m* обим датог скупа података, *tr* означава траг матрице, а *B* и *W* су матрице дисперзије између кластера, односно унутар кластера. Траг матрице *B* и *W* се рачуна као:

$$tr(B) = \sum_{i=1}^{k} |A^i| (z_i - z)^T (z_i - z),$$

$$tr(W) = \sum_{i=1}^{k} \sum_{x \in A^i} (x - z_i)^T (x - z_i),$$

где је *z* аритметичка средина (центроида) целог скупа, а z_i центроида кластера A^i. Максимална вредност овог индекса се користи за избор најбоље поделе.

Gamma индекс, као друга од мера интерне валидности, може се израчунати према следећој формули [111]:

$$G = \frac{S_+ - S_-}{S_+ + S_-} \epsilon [-1,1]$$

У датој формули S_+ представља број конкордантних парова објеката, док S_- представља број дискордантних парова и објеката и могу се дефинисати на следећи начин:

$$S_+ = \frac{1}{2}\sum_{l=1}^{k} \sum_{\substack{x_i, x_j \in A^l \\ x_i \neq x_j}} \frac{1}{2}\sum_{m=1}^{k} \sum_{\substack{x_p \in A^m \\ x_q \notin A^m}} d\left(\|x_i - x_j\| < \|x_p - x_q\|\right),$$

$$S_- = \frac{1}{2}\sum_{l=1}^{k} \sum_{\substack{x_i, x_j \in A^l \\ x_i \neq x_j}} \frac{1}{2}\sum_{m=1}^{k} \sum_{\substack{x_p \in A^m \\ x_q \notin A^m}} d\left(\|x_i - x_j\| > \|x_p - x_q\|\right)$$

Пар растојања (различитости) је конкордантан (дисконкордантан), уколико је растојање унутар кластера стриктно мање (стриктно веће) него растојање између кластера. Боља подела се очекује за веће вредности S_+, мање вредности S_-, односно веће вредности индекса *G*.

Silhouette индекс се користи за оцењивање отималног броја кластера у подацима [112]. Уколико се са $X = \{x_1, x_2, \dots, x_m\}$ означи скуп од m објеката груписаних у k кластера $A^1, \dots, A^k$. Нека је $A^j = \left\{x_1^j, x_2^j, \dots, x_{m_j}^j\right\}$ j-ти кластер, $j=1,\dots,k$, где је $\left|A^j\right| = m_j$. Уколико се са $d(x_i^j, x_s^j)$ означи растојање између i-тог објекта из кластера A^j и s-тог објекта у истом кластеру прво просечно растојање a_i^j између i-tog објекта из кластера A^j и свих других објеката у истом кластеру може се дефинисати као:

$$a_i^j = \frac{1}{m_j - 1}\sum_{s=1}^{m_j} d(x_i^j, x_s^j), \quad i = 1, \dots, m_j$$

Минимално просечно растојање између *i*-тог објекта у кластеру A^j и свих других објеката у кластеру A^s, $s=1,\ldots,k$, $s \neq j$ је дефинисано на следећи начин:

$$b_i^j = \min_{\substack{l=1,\ldots,k \\ l \neq j}} \left\{ \frac{1}{m_l} \sum_{s=1}^{m_l} d(x_i^j, x_s^l) \right\}, \qquad i = 1, \ldots, m_j$$

Silhouette ширина *i*-тог објекта, који припада *j*-том кластеру A^j, се рачуна као:

$$s_i^j = \frac{b_i^j - a_i^j}{max\{a_i^j, b_i^j\}} \in [-1,1]$$

Silhouette кластер A^j се дефинише као:

$$S_j = \frac{1}{m_j} \sum_{i=1}^{m_j} s_i^j$$

На основу свега наведеног, *silhouette* индекс се може дефинисати као просечна silhouette ширина за све објекте у датом скупу:

$$S = \frac{1}{k} \sum_{j=1}^{k} S_j$$

Овај индекс одражава компактност података унутар кластера и раздвојеност између различитих кастера. Вредност ове мере налази се у интервалу $[-1,1]$. Оптимална вредност броја кластера *к* бира се тако да максимизира вредност *S*.

Бајесов информациони критеријум спада у меру интерне валидности кластеризације. Максимизирање логаритма функције веродостојности је еквивалентно минимизирању Бајесовог информационог критеријума. За два дата модела уклапања који се примењују на исти скуп података, модел са мањом вредношћу информационог критеријума се сматра бољим [113].

Проблеми са којима се сређу истраживачи у кластер анализи су велики број показатеља сличности (растојања), велики број метода, одређивање скупа релевантних променљивих, недостајући подаци, одређивање оптималног броја

кластера и валидност решења [78]. Значајан изазов у кластер анализи представља рад са великим скуповима података и великим бројем обележја, посебно рад са категоријалним, односно комбинованим типовима обележја. Важно је напоменути да многи од проблема повезаних са кластер анализом представљају проблеме у мултиваријантној статистици, а то су: избор одговарајуће мере, избор променљивих, унакрсна валидација и екстерна валидност. Одређивање скупа релевантних променљивих је као и код већине мултиваријантних метода једна од најважнијих одлука, јер сама техника кластер анализе не разликује релевантне од нерелевантних променљивих. Што је више променљивих укључено у кластер анализу и што су оне више међусобно независне, теже је пронаћи одговарајући образац за груписање јединица посматрања [109]. Укључивање једне ирелевантне променљиве повећава вероватноћу утицаја *outlier*-а, што може значајно да утиче на резултате. Мора се водити рачуна о мултиколинеарности променљивих. Већина алгоритама кластеровања је ограничена на рад са скуповима података који садрже непрекидна обележја. Међутим, у реалним ситуацијама често су присутни велики скупове података са категоријалним и комбинованим типовима обележја. Ово представља велики изазов са математичког становишта, у смислу креирања новог ефикасног приступа у кластеровању оваквих података. Неке од алтернативних метода валидације кластеровања су индекси засновани на хомогености и/или сепарацији, поређење различитих метода кластеровања на истом скупу података, визуелна валидација кластера, тестови хомогености скупа у поређењу са алтернативним кластеровањем и коришћење екстерних информација. Један од начина процене валидности кластерских решења обухвата тестирање разлика између кластера на променљивама коришћеним у поступку кластер анализе. Овај приступ подразумева коришћење различитих статистичких техника, у зависности од врсте и броја обележја, као и броја кластера. Међутим, недостатак оваквог приступа представља то што објекти нису сврстани у кластере по случају, већ на основу максимизирања растојања између кластера по коришћеним променљивама.

5. Прикупљање метеоролошких и просторно-временских података

Прецизна пољопривреда се у основи заснива на праћењу тренутних метеоролошких параметара и креирању што је могуће прецизније предикције засноване на метеоролошким параметрима. У зависности од метеоролошких услова могу се разликовати дневне активности као и активности које ће се обављати у неком будућем периоду. Посматрано из угла метеоролошких промена, праћење метеорлошких услова настаје као спој примене науке и технологије у циљу предикције стања атмосфере за дату област посматрања [12]. Коришћењем система за праћење метеоролошких промена агрономи и пољопривредни произвођачи могу прикупити информације о различитим временским параметрима, као што су: температура ваздуха, влажност ваздуха, брзина и смер ветра, количина падавина итд. Поред поменутих метеорлошких параметара, метеоролошке станице могу вршити и мерење специфичних параметара, као што су влажност земљишта, влажност листа биљке итд. На основу тренутних мерења метеоролошких параметара и мерања вршених током дужег временског периода у прошлости, могу се креирати прецизни модели извршења пољопривредних активности. Хемијска заштита гајених биљака директно је наслоњена на познавање метеоролошких параметара, па самим тим представља једну од главних активности пољопривредних произвођача условљену метеоролошким условима. Како би прикупљање метеоролошких параметара са производних површина допринело примени истих у процесу доношења одлука везаних за хемијску заштиту, метеоролошке станице морају бити стратешки распоређење. Правилно распоређивање метеоролошких станица повезано је са конфигурацијом терена, опсегом покривености и сврхом саме станице. Поред позиције метеоролошке станице на производној површини овакви системи морају задовољити низ других техничких захтева.

Различити типови метеоролошких станица, посебно аутоматизованих, користе се за прикупљање потребних података. Када се говори о опремљености метеоролошке станице у циљу адекватног коришћења свака од метеоролошких станица требало би да буде опремљења одговарајућом групом сензора. Скуп

сензора којима ће конкретна метеорлошка станица бити опремљена зависи од унапред дефинисане намене. Просторно-временски параметри поред метеоролошких параметара чине још једну од битнијих компоненти у раду метеорлошке станице. Значајност просторно-временских параметара посебно се огледа у ситуацијама када на одређеном локалитету постоји систем од више метеоролошких станица. Овакав систем у је највећем броју случајева повезан са једном јединственом базном станицом до које се врши слање измерених вредности метеоролошких параметара. Просторно-временска компонента сваког од пакета података уводи могућност кластеризовања података на основу локације на којој су измерене дате вредности, као и на основу времена када је мерење извршено. На овакав начин је, уколико се на различитим локалитетима узгајају различите културе, омогућено прецизније сагледавање метеоролошке ситуације посматрањем искључиво метеоролошких мерења за дати локалитет. Самим тим, приликом креирања предикционих модела за предикцију времена хемијских третмана на бази метеоролошких параметара за конкретан локалитет може се избећи непотребна обрада параметара који припадају неком другом локалитету. Референцирањем података на основу просторно-временске компоненте поред смањења скупа података који је потребно обрадити, у великој мери утиче на отклањање шума који би се у подацима јавио услед присуства мерења са другог локалитета, што доводи до креирања прецизнијег модела.

Поред одабира одговарајуће групе сензора и реализације преноса пакета података додатни лимитирајући фактор је и напајање свих компоненти метеоролошке станице. У највећем броју случајева напајање метеоролошке станице намењене монтажи на пољопривредним површинама у условима у којима не постоји дистрибутивна мрежа електричне енергије врши се помоћу фотонапонских панела (соларних панела). Како је реализација метеоролошке станице у великој мери условљена и ценом компоненти доступних на тржишту, када се говори о компонентама које би се могле искористити у циљу реализације сопствене метеоролошке станице готово све компоненте је могуће купити на слободном тржишту. На овакав начин креирање сопствене метеоролошке станице може бити знатно повољније од куповине комерцијале станице неког од познатих произвођача.

Измерене вредности метеоролошких и просторно-временских параметара на неком удаљеном локалитету не значе пуно уколико се очитавање истих мора обавити на самом локалитету. Како се метеоролошке станице намењене употреби у пољопривреди налазе на производним површинама које су удаљене од места становања лица које прати промену параметара исте би требало да поседују могућност слања података до базне станице. Практично, поред опремљености метеоролошке станице потребним сензорима она мора бити опремљена и одговарајућим хардвером којим ће омогућити слање података. Успешно слање пакета података не сме бити условљено растојањем између метеоролошке и базне станице.

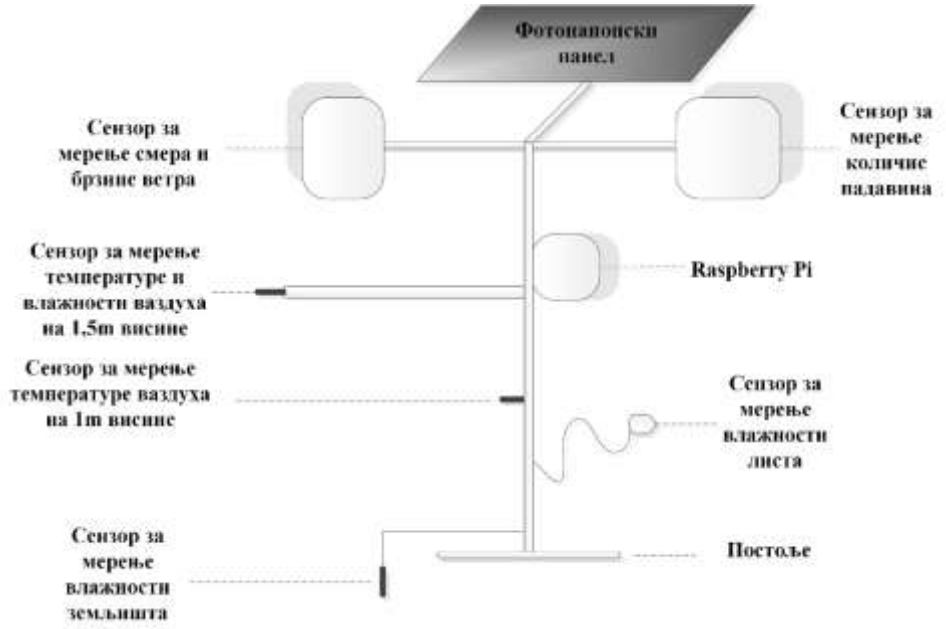
У наставку овог поглавља дат је предлог модела метеоролошке станице. Предложени модел креиран је на бази опреме доступне на тржишту са освртом на практичну ефикасност и економске издатке. У циљу одабира најадекватнијих компоненти извршена је анализа компоненти доступних на тржишту. Анализиране су карактеристике компоненти прописане од стране произвођача и тржишна цена истих. Развој датог модела је подстакнут потребама предикције времена хемијских третмана која је условљена тачношћу метеоролошких података добијених са метеоролошке станице. Опис предложеног модела метеоролошке санице дат је кроз опис метеоролошких компонената, просторно-временских компонената и компонената потребних за реализацију напајања.

5.1 Креирање модела метеоролошке станице

Шематска организација предложеног модела метеоролошке станице и распоред компоненти приказани су на слици 4. Приказано постоље и стуб метеорлошке станице користе се као основа за монтажу рачунара, мерних сензора и уређаја, као и напајања саме станице. Постоље и стуб треба да обезбеде довољну стабилност и одговоре утицају различитих метеоролошких услова којима би метеоролошка станица могла бити изложена. Централни део на стубу метеоролошке станице заузима *Raspberry Pi 3 Model B* рачунар. *Raspberry Pi* рачунар представља главну компоненту метеоролошке станице и управља радом свих сензора[1]. Поменути

[1] Raspberry Pi, Raspberry Pi Products, https://www.raspberrypi.org/products/, datum pristupa: 15.10.2017.

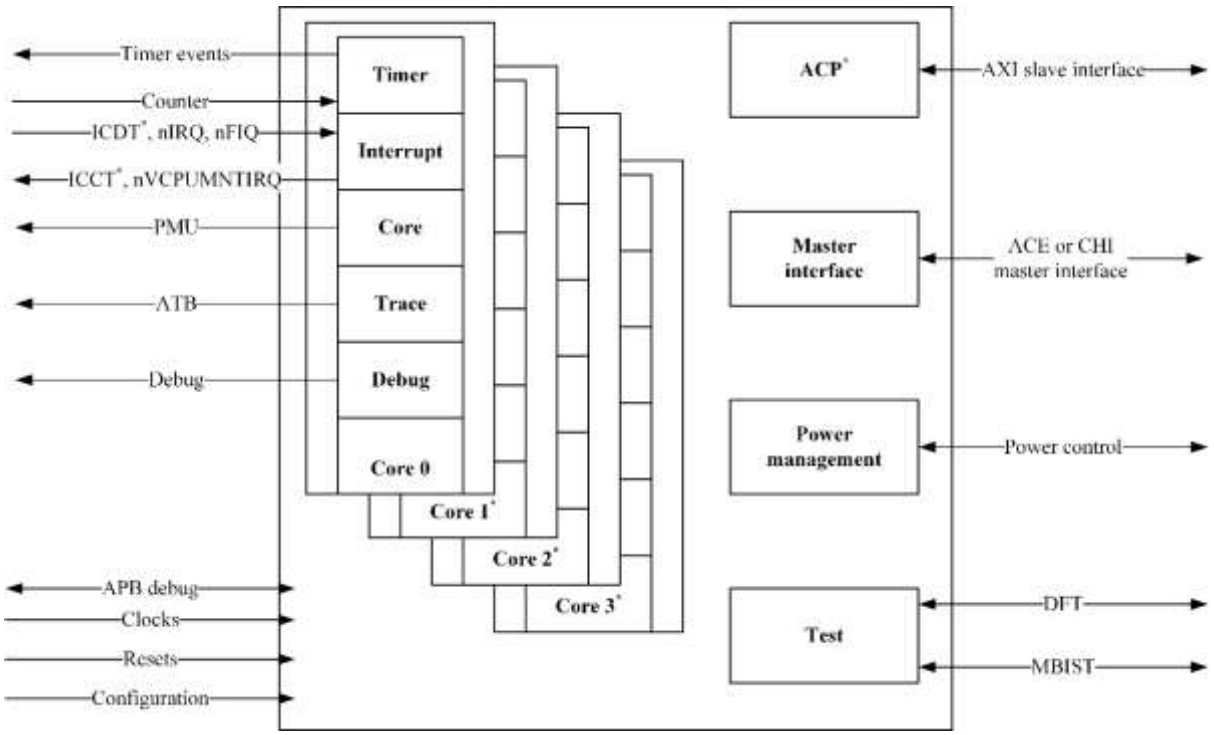
Raspberry Pi уређај представља трећу генерацију овог уређаја. На слици 5. приказан је *Raspberry Pi* уређај у основној верзији, без додатних компоненти. *Raspberry Pi* је рачунар малих димензија са оперативним системом на *Micro SD* меморијској картици. Поред малих димензија и ниске цене, карактеристике овог рачунара не заостају за карактеристикама персоналних рачунара.



Слика 4: Шематска организација компоненти метеоролошке станице

Слика 5: Изглед Raspberry Pi плоче треће генерације, слика преузета са https://www.raspberrypi.org/products/raspberry-pi-3-model-b/

Карактеристично за трећу генерацију овог рачунара је и то да ради на *GNU/Linux* платформи као и на *Windows 10* платформи. На овакав начин даје добру основу за евентуално проширење и комуникацију са другим уређајима. Рад овог рачунара је заснован на *Broadcom BCM2837B0, ARM Cortex-A53* 64-битном процесору брзине од 1.4GHz. *Cortex-A53* је процесор средњег ранга, кога карактерише мала потрошња енергије и који имплементира *Armv8-A* архитектуру. Овај процесор заснива се на могућа четири језгра, при чему свако од језгара садржи сопствену *L1* меморију. Поред *L1* меморије процесор поседује и једну дељиву *L2* меморију. Пример *Cortex-A53 MPCore* конфигурације са четири језгра и *ACE* или *CHI* интерфејсом дат је на слици 6.

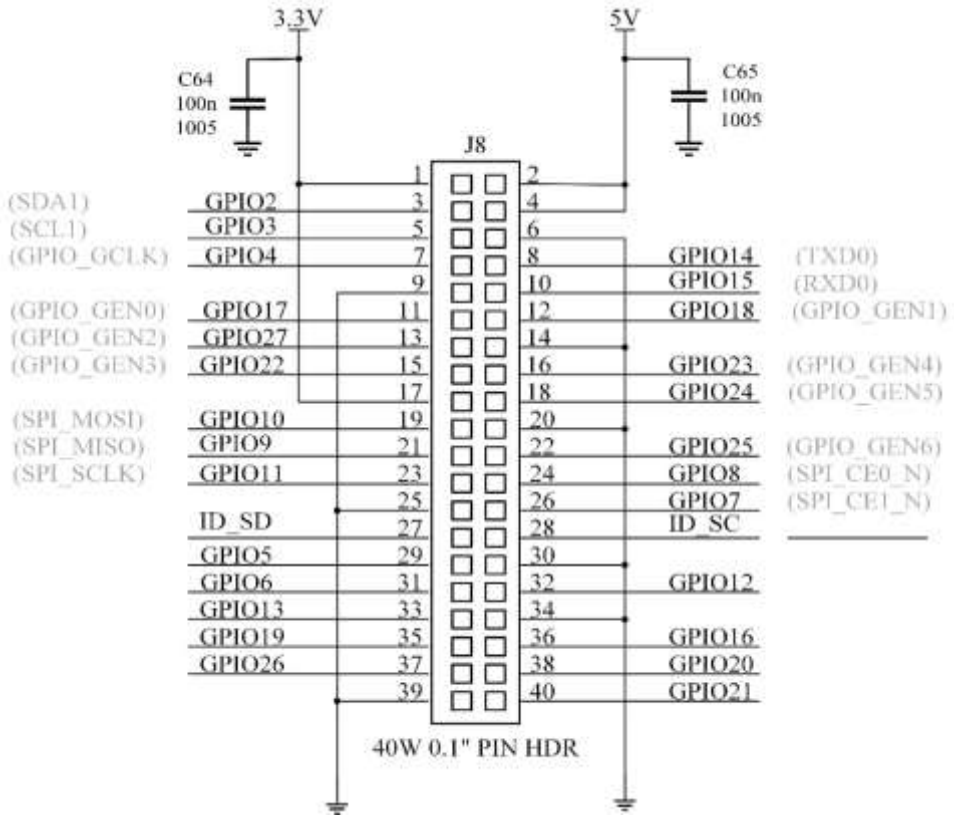


Слика 6: Пример конфигурације Cortex-A53 процесора

Компоненте овог процесора означене звездицом представљају опционе компоненте. На основу датих ознака на слици 6. може се закључити да су јездра 1-3 опциона, што значи да према овој архитектури процесор може имати само једно језгро, а то је јездро 0, док је хардверски дата могућност проширења до максималних четири. Поред језгара, као додатни сигнали, могу бити уведени *stream* протокол сигнали од дистрибутора до интерфејса, као и од интерфејса до дистрибутора у ознаци *ICDT* и *ICCT* респективно. Такође, опционим се сматра

ACP port (енг. *Accelerator Coherency Port*). Овај процесор имплементира *Armv8-A* архитектуру, која укључује подршку за *AArch32* и *Aarch64* стања извршења. Исто тако, поменута архитектура обезбеђује подршку за све нивое извршења *EL0, EL1, EL2,* и *EL3* у сваком од стања извршења. Поред поменуте подршке архитектура укључује *A32, A64* и *T32* сет инструкција. *Cortex-A53* процесор има један улаз такта, те се сва језгра и *SCU* (енг. *Snoop Contol Unit)* тактују дистрибуираном верзијом такта у ознаци *CLKIN*.

Повезивање сензора са овим рачунаром обавља се преко улазно излазних пинова опште намене (енг. *General Purpose Input/Output pins - GPIO*) који се налазе на плочи рачунара и којих има укупно четрдесет. Шематски приказ *GPIO* пинова дат је на слици 7.



Слика 7: Шематска организација GPIO пинова

GPIO пинови могу се конфигурисати као улазни пинови опште намене, као излазни пинови опште намене или као једна од шест алтерантивних специјалних намена. Свих четрдесет *GPIO* су подељени у три банке у складу са поменутим процесором. Свака од банака има свој *VDD* улазни пин.

Такође, свака од банака се напаја са 3,3V. Ово практично значи да би повезивање *GPIO* пинова на већи напон проузроковало оштећење *GPIO* блока. Од укупног броја пинова, два пина пружају напајање од 5V, док, такође, два пина пружају напајање од 3,3V. У исто време на плочи се налази већи број пинова уземљења који нису конфигурисани. Сви остали пинови су, као што је наведено, пинови опште намене чији је излаз постављен на 3,3V, док је улаз толерантан до 3,3V. Ово практично значи да се пинови који су дизајнирани као излазни могу поставити на напајање до 3,3V или мање.

Излазни *GPIO* пинови се могу читати као високи напон до 3,3V или ниже до 0V. Ово се омогућава коришћењем унутрашњег *pull-up* или *pull-down* отпорника. *GPIO2* и *GPIO3* пинови имају фиксне *pull-up* отпорнике, док се за остале пинове конфигурација може извршити софтверски. Сваки од *GPIO* пинова који је конфигурисан као улазни пин опште намене може се конфигурисати и као извор прекида за *ARM*. За неколико генерација извора прекида може се конфигурисати:

- ниво осетљивости (висок/низак);
- растућа/опадајућа ивица такта;
- асихроно растућа/опадајућа ивица такта.

Прекиди базирани на дефинисаном нивоу осетљивости одржавају исти статус све док системски софтвер не обрише креирани ниво, и то ресетовањем периферсног уређаја који је генерисао прекид. У случају прекида генерисаног растућом или опадајућом ивицом мала количина синхронизације је уграђена у процес детекције. На системској фреквенцији такта пин је узоркован критеријумом за генерисање прекида који представља стабилан прелаз у прозору од три циклуса. Са друге стране, асихроно детектовање заобилази поменуту синхронизацију како би се омогућило откривање веома малих догађаја. Поред поменутих намена *GPIO* пинови могу имати и алтернативне функције, од којих су неке доступне на свим пиновима док су поједине доступне само на специфичним пиновима. С тим у вези, могу се издвојити функције као што су *PWM* (енг. *Pulse-Width Modulation*), *SPI*, *I2C*, и *Serial*. Поменуте функције се са једне стране могу дефинисати софтверски на свим или на одређеној групи пинова, док су хардверски доступне на тачно одређеним пиновима. На пример, софтверски *PWM*

се може дефинисати на свим пиновима док је хардверски доступан на пиновима *GPIO12, GPIO13, GPIO18,* и *GPIO19*. На самој плочи постоје и пинови који су резеривисани. Тако су пинови *ID_SD* и *ID_SC* дефинисани за *HAT* и *EEPROM* респективно. Практично током бутовања *I2C* интерфејс ће претраживати постојање *EEPROM* меморије која идентификује повезану плочу и омогућава аутоматско подешавање *GPIO* периферије и опционо *Linux* драјвера.

Поред улазно излазних пинова опште намене *Raspberry Pi 3* опремљен је још и *1GB LPDDR2 SDRAM*-ом. Што се тиче излазних портова за репродукцију слике и звука, опремљен је са по једним *HDMI* портом, *MIPI DSI display* портом и *MIPI CSI* камера портом, као и са једним четворополним стерео излазним и композитним видео портом. Самим тим, по питању мултимедијалне репродукције подржава следеће формате: *H.264, MPEG-4* декодер; *H.264* енкодер, *OpenGL ES 1.1, 2.0* графику. Додатна могућност повезивања овог уређаја са другим уређајима може се остварити путем бежичне везе и четири *USB 2.0* порта. Што се тиче бежичне везе, уређај је опремљен чипом који омогућава успоставу бежичне везе на фреквенцији од 2.4 GHz, као и 5 GHz IEEE 802.11.b/g/n/ac b. Поред ове две могућности поседује и могућност повезивања помоћу *bluetooth*-а 4.2. Такође *Rasperry Pi* уређај може се конектовати на мрежу помоћу *Gigabit Ethernet* порта, што омогућава максималну брзину преноса података од 300Mbps. Што се тиче напајања, *Raspberry Pi* рачунар се напаја помоћу микро *USB* конектора снаге 5V/2.5A DC. Такође, уређај се може напајати преко улазно излазних пинова опште намене, као у путем *Ethernet* порта (енг. *Power Over Ethernet - PoE*). Опсег температура окружења у којој *Raspberry Pi* може радити износи 0-50ºC. Опремљеност *Raspberry Pi* уређаја *USB* портовима, као и другим видовима конекције омогућава погодности у домену повезивања уређаја и проширења могућности овог рачунара. Када се ради о метеоролошкој станици за приступ *Rasbperry Pi* уређају на самој локацији ради подешавања могу се користити миш и тастатура који се екстерно могу повезати преко *USB* порта. Уређај поседује и могућност повезивања LCD дисплеја осетљивог на додир који омогућава очитавање временских параметара на самој станици, као и алтернативни приступ уређају ради подешавања.

Температурни сензор, као што се може видети са слике 4., позициониран је на 1m висине од земљине површине. Поред овог температурног сензора, у оквиру модела, дефинисан је још један сензор који ће вршити температурна мерења. Овај сензор је комбиновани сензор чија је улога да поред мерења температуре ваздуха врши и мерење релативне влажности ваздуха. Поменути комбиновани сензор постављен је на 1,5 m висине. Сензор за мерење влажности земљишта налази се на 20 cm испод површине земље и повезан је са *Raspberry Pi* уређајем помоћу кабла. Сензор који је намењен мерењу влажности листа није фиксиран за сами стуб метеоролошке станице већ се поставља у оквиру лисне масе гајене културе. Повезан је са *Raspberry Pi* рачунаром помоћу кабла, као што је био случај са сензором за мерење влажности ваздуха. На самом врху стуба метеоролошке станице налазе се сензори за мерење брзине и смера ветра. Позиционирање на самом врху елиминише могућност промене смера или брзине ветра услед утицаја механичког заклона насталог од неког другог дела станице. Последњи у низу сензора јесте сензор за мерење количине падавина који је помоћу сопственог носача причвршћен на стуб метеоролошке станцице. Такође, на врху стуба се налази соларни панел намењен напајању целокупног система. Како би се обезбедило конзистентно напајање и у зимским месецима, са мало сунчаних сати током дана, на самом постољу станице налазе се батерије које се пуне током сунчаних сати. Сви одабрани сензори се повезују са овим рачунаром и добијају напајање преко пинова за напајање екстерних уређаја. Идентификација сензора приликом имплементације комуникације са *Raspberry Pi* уређајем обавља се на основу јединственог серијског броја сваког од сензора [114].

5.2 Компоненте за прикупљање метеролошких параметара

Предложени модел метеоролошке станице намењене употреби у пољопривреди, као што је наведено, опремљен је одговарајућом групом метеоролошких сензора. У ову групу сензора могу се уврстити сензори за мерење: температуре ваздуха, влажности ваздуха, количине падавина, брзине и смера ветра. Поред сензора за праћење метеоролошких промена атмосфере, предложени модел станице опремљен је и сензорима специфичне намене, као што су сензори за мерење влажности листа и мерење влажности земљишта. Намена свих поменутих сензора је прикупљање адекватног скупа података на основу кога је

могуће касније доношење одлука. Како се на тржишту могу пронаћи потребни сензори различитих произвођача, а самим тим и различитих техничких могућности, извршена је анализа већег број сензора у циљу одабира одговарајућег. За потребе анализе дефинисани су општи критеријуми у погледу могућности повезивања са *Raspberry Pi* уређајем, као и односа квалитета и цене. Поред општих критријума који се примењују на сваки од сензора, дефинисани су посебни критеријуми за сваку од група сензора. Примера ради, за одабран скуп температурних сензора дефинисани су критеријуми специфични са ове сензоре као што је опсег мерења, водоотпорност и уторшак енергије, док су за неку другу групу сензора специфични други критеријуми. Код одабира сензора водило се рачуна и о степену грешке при мерењу прописаном од стране произвођача. На основу прописаних техничких карактеристика посматраних сензора, њихове цене и потреба будуће реализације извршен је одабир најадекватнијег сензора из сваке од група.

5.2.1 Температурни сензор

Познато је да је једна од карактеристика темпратуре ваздуха њена осцилација током дана. Како су поједини пољопривредни процеси зависни од дневних температура веома је важан и одабир одговарајућег температурног сензора. Анализа температурних сензора на основу карактеристика датих од стране произвођача приказана је у табели 1.

Табела 1: Поређење карактеристика различитих температурних сензора

Ознака сензора	Мерни опсег [ºC]	Прописана тачност [ºC]	Напајање [V]	Приближна цена [дин]
DS18B20	-55 do +125	± 0,5	3,0 – 5,5	1200,00
DS18B20-H	-55 do +125	± 0,5	3,0 – 5,5	1800,00
DS1818-10	-40 do +85	± 8,5	3,3	240,00
DS1822	-55 do +125	± 2,0	3,3 – 5,5	780,00
DS18B20Z	-55 do +125	± 0,5	3,0 – 5,5	768,00

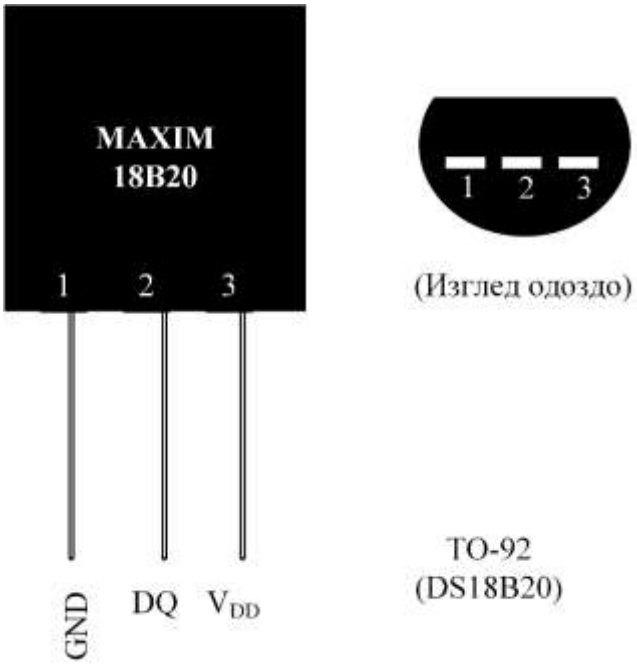
За мерење спољашње температуре из описане групе сензора одабран је DS18B20 температурни сензор. Ово је температурни сензор који долази постављен на крају кабла дужине 91 cm и пречника 4 mm.

Дужина кабла омогућава монтажу сензора на одређеној удаљености од *Raspberry Pi* уређаја. Такође, сами сензор је водоотпоран. Због начина повезивања као и своје водоотпорности савршен је за примену у спољашњим условима. На слици 8. може се видети изглед самог сензора, док је на слици 9. дат шематски приказ сензора.

Слика 8: Изглед DS18B20 температурног сензора, слика преузета са: https://opencircuit.shop/Product/12661/Crowtail-DS18B20-One-Wire-Waterproof-Temperature-Sensor

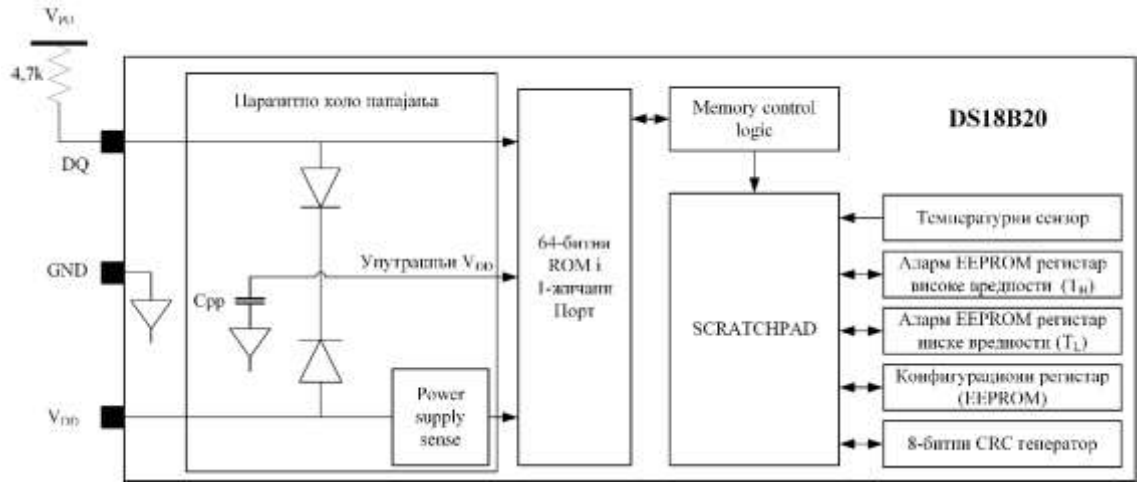
С обзиром на то да се ради о дигиталном сензору, додатна повољност је пренос сигнала на већем растојању без деградације истог. Још једна од повољности јесте повезивање са *Raspberry Pi* рачунаром. Наиме за повезивање и комуникацију потребан је само један дигитални пин, што оставља простора за повезивање других сензора или више истих температурних сензора на различитим позицијама. Управо из овог разлога, сваки од *DS18B20* сензора има јединствени 64-битни *ID* креиран од стране произвођача, што омогућава повезивање више сензора на један улазно излазни пин опште намене. Одзив приликом очитавања је мањи од 750 ms и одговара конверзији измерене аналогне верије температуре 12-битној бинарној речи. Приписана тачност од ± 0,5 ºC одговара мером опсегу од -10 ºC до +85 ºC. Уколико се мерење врши у опсегу између наведеног и максималног, тачност може бити мања. Одабрана верзија овог сензора јесте *TO-92* пакет са конекцијом путем 3 пина, као што се може видети на слици 9.

Поређењем распореда пинова на слици 9 и жичане везе овог сезора дате на слици 8. долази се до информације о томе да црвена жица заправо представља пин 3, црна пин 1, а бела пин 2. Пин 3 представља додатну могућност напајања, што значи да уколико се напајање врши преко линије за податке, тачније преко пина 2, овај пин мора бити повезан са уземљењем. Оваква могућност значи да практично нема потребе за додатним извором напајања. Пин 2 у ознаци *DQ* као улазно излазни пин података повезује се са дигиталним улазно излазним пином опште намене на *Raspberry Pi* уређају.



Слика 9: Распоред пинова DS18B20 температурног сензора

Блок дијаграм DS18B20 сензора приказан је на слици 10. S*cratchpad* моморија састоји се од регистра величине 2 бајта у коме се чува дигитална вредности температуре добијене са сензора. Додатно, ова меморија може обезбедити приступ регистру величине 1 бајта намењеном за потребе окидача за горњу и доњу границу температуре у ознаци T_H, T_L респективно, као и конфигурационом регистру величине од 1 бајт. Конфигурациони регистар омогућава корисницима подешавање резолуције конверзије температуре у дигитални облик на 9, 10, 11 или 12 битова. T_H, T_L и конфигурациони регистри представљају *EEPROM* меморију, што значи да чувају податке и након нестанка напајања, те самим тим њихово конфигурисање није потребно са сваким покретањем уређаја.



Слика 10: Блок дијаграм DS18B20 температурног сензора

Поменута конфигурабилност температурног сензора на 9, 10, 11 или 12 битова одговара инкременту од 0,5 ºC, 0,25 ºC, 0,125 ºC, и 0,0625 ºC респективно. Фабричка резолуција овог сензора подешена је на 12 битова. Пратећи овакву конверзију температурни подаци се након мерења памте у температурном регистру величине 2 бајта, након чега се сензор враћа у стање *idle*. Овај регистар се налази у *scratchpad* меморији. У зависности од тога како је извршено напајање овог сензора, постоји могућност креирања система нотификације, при чему ће након прослеђивања команде за конверзију *DS18B20* сензор одговорити прослеђивањем 0 током трајања конверзије и прослеђивањем 1 уколико је конверзија готова. Уколико се напајање врши преко линије за податке, оваква нотификација није могућа, док је иста могућа уколико се напаја преко спољашњег напајања. Калибрација излазне вредности температуре извршена је у степенима целзијусовим што одговара подручју на коме је рађено ово истраживање. Формат температурног регистра дат је на слици 11.

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}

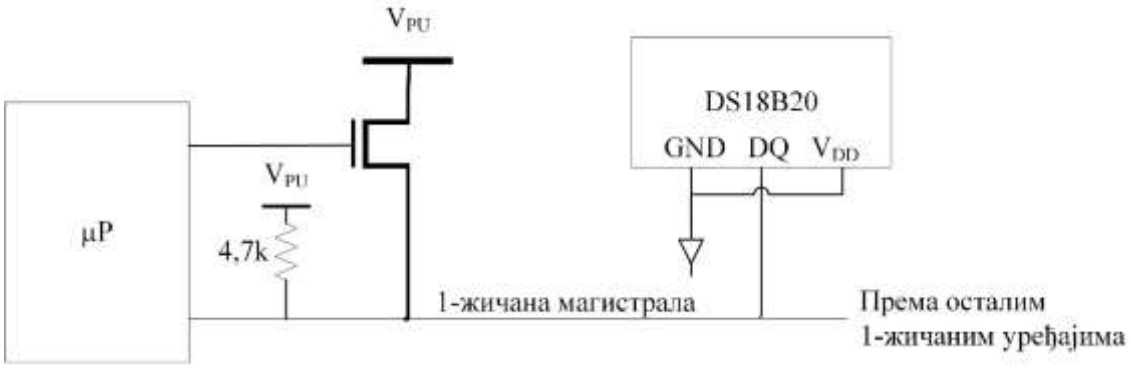
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

S – Знак

Слика 11: Формат регистра температурног сензора

Према овом формату температурни подаци се смештају у 16-битни означени комплемент двојке у оквиру температурног регистра. Битови знака који су на слици 11. означени са *S* користе се као индикатори да ли је температура позитивна или негативна. Уколико је измерена вредност температуре позитивна бит знака ће добити вредност C=0. Са друге стране, уколико је вредност температуре испод нуле, вредност бита знака ће износити C=1. Што значи да, ако је *DS18B20* сензор конфигурисан тако да измерене вредности конвертује у 12 бита, сви битови ће у температурном регистру садржати валидне податке. Уколико је конфигурација таква да се користи 11 битова, бит 0 ће бити недефинисан. За регистар са 10 битова, битови 1 и 0 ће бити недефинисатни. На основу оваквог начина дефинисања битова, уколико се ради о регистру са 9 битова, битови 2, 1 и 0 остаће недефинисани.

Као што је раније наведено, напајање *DS18B20* се може обавити на два начина, помоћу екстерног V_{DD} пина или може радити у такозваном паразитном моду у коме ће се напајати преко линије за податке. Како је за предлог модела метеоролошке станице одабрано напајање преко паразитног мода на слици 12. дата је шема оваквог начина напајања.



Слика 12: Напајање DS18B20 сензора током температурне конверзије у паразитном моду

Предности паразитног мода напајања огледају се у примени температурног сензора за мерење температуре путем даљинског очитавања, као и у случајвима када је мерну опрему потребно упаковати на малом простору. Овакво повезивање омогућава напајање *DS18B20* сензора све док је линија за податке активна. Из тог разлога се уводи кондензатор који служи за чување енергије која се користи за напајање овог сензора у времену када линија за податке није активна. У сваком

случају, када овај температурни сензор врши конверзију температуре или копирање вредности температуре из *scratchpad* меморије у *EEPROM* меморију, оптерећење које се ствара може износити до 1.5 mA. Овакав утрошак може довести до неприхватљивог пада напона дуж слабог једнолинијског отпорника, што је уједно и више струје него што може да се испоручи путем *Cpp*. Како би се обезбедио неометан рад система потребно је обезбедити овако јако повлачење енергије изнова и изнова приликом сваке конверзије температуре или приликом сваког пребацивања из једне у другу меморију. Ово се може обезбедити коришћењем *MOSFET*-а како и би се напајање у датим моментима обезбедило директно са извора напајања. Овакво напајање би трајало максимално 10 μs у тачно предифинисано време након конверзије или након копирања из меморије у меморију. Током трајања ових двеју операција и током трајања оваквог вида напајања остале операције на магистрали се прекидају, тако да иста остаје доступна само овом сензору. Према техничкој документацији овог темпратурног сензора паразитно напајање не би се требало користити уколико температура окружења превазилази +100 ºC. На оваквим температурама *DS18B20* сензор није у могућности да одржи комуникацију због већих струја цурења које се могу јавити. У таквим случајевима напајање треба обавити преко посебне линије за напајање.

5.2.2 Сензор за мерење температуре и релативне влажности ваздуха

Релативна влажност ваздуха још један је од метеоролошких параметара за чије праћење је потребно одабрати одговарајући сензор. Сензори за мерење релативне влажности ваздуха најчешће се на тржишту налазе у комбинацији са температурним сензорима. Наиме, у оквиру једног сензора врши се комбиновано мерење два параметра. Управо из овог разлога је вршено поређене перформанси најзаступљенијих комбинованих сензора за мерење температуре и релативне влажности ваздуха. Резултати анализе поређења доступних сензора приказани су у табели 2.

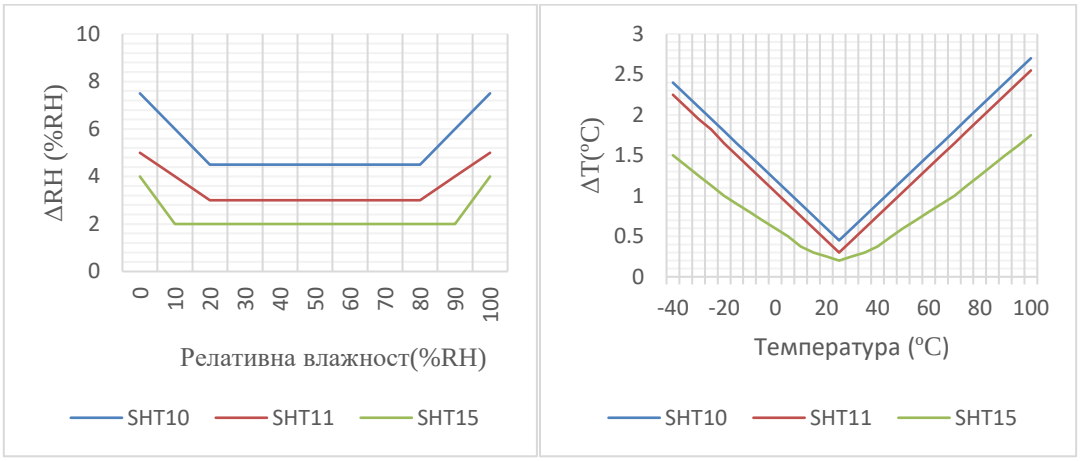
Поред приказаних разлика између наведених сензора које се односе на мерне опсеге влажности ваздуха и температуре, као и прописане тачности приликом мерења, још једна од карактеристика, као што је био случај са температурним сензором, јесте отпорност на временске услове.

Табела 2: Поређење карактеристика различитих сензора за мерење релативне влажности ваздуха

	Релативна влажност		Температура			
Ознака сензора	Мерни опсег [%]	Прописана тачност [%]	Мерни опсег [ºC]	Прописана тачност [ºC]	Напајање [V]	Цена [din]
SHT15	0 - 100	±2.0	-40 do +120	±0,3	2,4 – 5,0	6000,00
HIH6030	0 - 100	±4,5	-40 do +100	± 0,5	2,3 – 5,5	800,00
DHT22	0 - 100	± 2-5	-40 do +80	± 0,5	3,0 – 5,0	1800,00
SHT10	0 - 100	±4,5	-40 do +120	± 0,5	2,4 – 5,0	6000,00
SHT11	0 - 100	±3.0	-40 do +120	±0,4	2,4 – 5,0	6000,00
AM2315	0 - 100	± 2,0	-20 do +80	± 0.1	3,5 – 5,5	3600,00
DHT11	20 – 80	± 5,0	0 do +50	± 2,0	3,0 – 5,0	600,00

Поред сличних техничких карактеристика којима се одликују, као и чињенице да су оба сензора отпорна на временске услове, чињеница која их раздваја јесте цена. *SHT1x* сензори, као што се може видети из Табеле 2 далеко су скупљи него што је то случај са *HIH6030* сензором. Међутим, детаљнијим поређењем и истраживањем утврђено је да *HIH6030* сензор није довољно отпоран на временске улове, односно да правилан рад овог сензора може ометати прашина и кондензација. Додатна улагања у заштиту овог сензора од временских услова, изведена тако да не утичу на његове перформансе, скоро да би изједначила цену овог сензора са ценом сензора из серије *SHTx*. Такође, чињеница да сензори серије *SHTx* долазе унапред заштићени са могућношћу повезивања путем кабла отвара додатне погодности у њиховом коришћењу. Како се у овој серији налазе три различита сензора по карактеристикама, извршен је одабир једног. Разлике између ова три сензора која припадају истој групи могу се видети на слици 13. Према техничким карактеристикама, *SHT15* сензор се одликује најпрецизнијим мерењима. Притом опесег мерења, напајање и цена овог сензора су истоветни као што је то случај са осталим сензорима ове серије. Уколико се са слике 13 упореде техничке карактеристике *SHT10, SHT11* и *SHT15* сензора поред међусобних разлика може се уочити и пад карактеритиска. Што се тиче одабраног *SHT15* сензора, приликом мерења релативне влажности ваздуха највећа грешка приликом мерења јавља се када је релативна влажност ваздуха у опсегу од 0-10% и у опсегу од 90-100%. У наведеним опсезима реалтивне влажности ваздуха

максимална вредност грешке може износити ±4%. Посматрајући карактеристику релативне влажности ваздуха преостала два типа сензора из ове серије *SHT10* и *SHT11* показују лошије перформансе. Катактеристике температурних мерења дате на слици 13. такође показују да *SHT15* сензор има најбоље перформансе. Најмању грешку приликом мерења овај сензор показује на температури ваздуха од 25 ºC.



Слика 13: Максимална толеранција релативне влажности ваздуха и температуре ваздуха за сваки од типова SHTx сензора

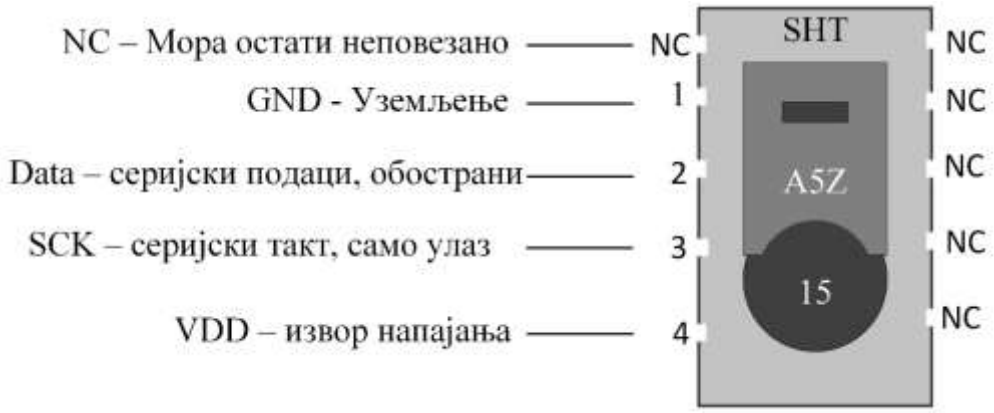
Изглед SHT15 сензора дат је на слици 14. На датој слици може се видети изглед овог сензора са и без заштитног дела. Заштитни део овог сензора креиран је тако да спечава улазак воде у оквиру мерног дела, те спречава могућност његовог оштећења. У исто време заштитни део омогућава пролазак ваздуха, чиме је обезбеђено мерење релативне влажности. Целокупно тело сензора заједно са заштитиним делом дугачно је 50 mm и пречника је 14 mm.

Слика 14: Изглед SHT15 сензора са и без заштитног дела, слика преузета са:

https://www.adafruit.com/product/1298

Такође, као што је раније наведено, овај сензор је опремљен каблом којим се врши повезивање са микроконтролером. Дужина овог кабла износи 1 m. Сами кабл је заштићен од утицаја временских услова и састоји се од 4 жице. Свака од жица има унапред дефинисану намену. Силиконски чип унутар заштитног дела поред сензора за мерење релативне влажности ваздуха и температуре вазуха садржи и појачавач, *AD* конвертор, *OTP* меморију и дигитални интерфејс.

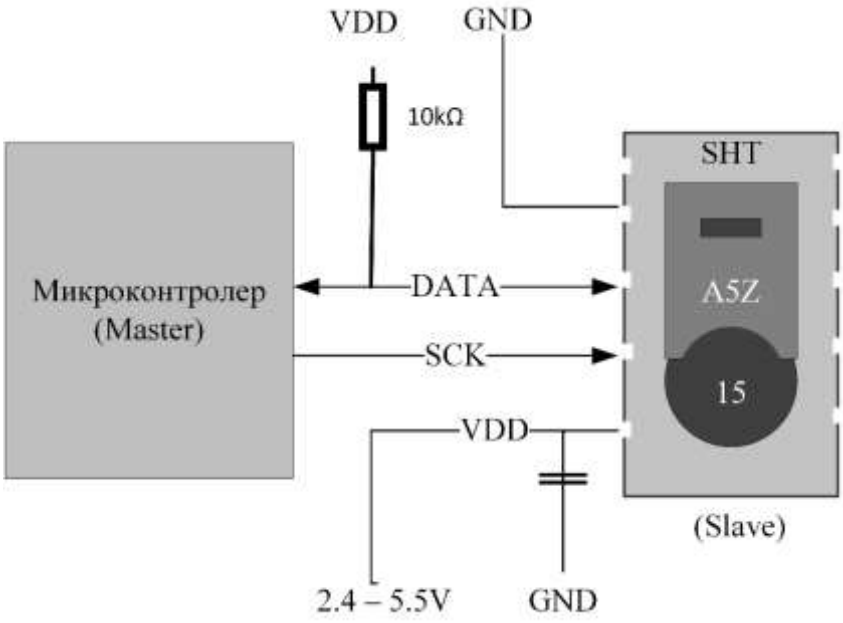
Распоред пинова *SHT10x* сензора може се видети на слици 15. Уколико се распоред пинова упореди са распоредом жица у каблу којим се сензор повезује са микроконтролером, може се видети да црвена жица одговара пину 4 који представља напајање, црна или зелена жица одговарају пину један који предствала уземљење, жута жица одговара пину 4 који представља такт и последња плава жица одговара пину 2 који представља пин за пренос података.



Слика 15: Распоред пинова SHT15 сензора

Напајање *SHT15* сензора, као што се на основу распореда пинова може видети, врши се преко линије за напајање. Оптималан рад сензора остварује се напајањем од 3,3 V. Погодност овог сензора је и мали утрошак енергије, с обзиром да поседује мод мировања, што значи да у периоду између два мерења овај сензор одлази у мод мировања, при чему утрошак енергије износи 2-5 μW енергије, док током периода активног рада има прописани утрошак енергије од 90 μW. Такође, утрошак енергије током процеса мерења релативне влажности ваздуха и температуре ваздуха износи 3 mW. Оваква функционалност посебно је значајна када се ради о имплементацији овог сензора у системе који нису повезани на стални извор напајања, већ се напајају путем батерија или соларног панела.

У неким случајевима мод одмора значајно може умањити потребу за капацитивношћу потребних батерија и соларног панела. Уколико се користи повезивање помоћу кабла пин напајања (*VDD*) и пин уземљења (*GND*), морају се раздвојити помоћу кондензатора од 100 nF, што се може видети на шеми кола *SHT15* сензора која је дата на слици 16.



Слика 16: Шема кола SHT15 сензора

Серијски интерфејс *SHT15* сензора је оптимизован за очитавање сензора и ефективан утрошак енергије. Када је у питању адресирање, овај сензор се не може адресирати помоћу I^2C протокола. Међутим, и поред тога што је адресирање помоћу овог протокола немогуће, сензор се може повезати на I^2C магистралу без интерференције са осталим уређајима који су повезани на исту магистралу. Приликом овакве реализације микроконтролер је задужен за прелазак између протокола. Такт сензора на слици 16. означен је са *SCK* и користи се за синхронизацију комуникације између микроконтролера и *SHT15* сензора. Линија за податке преко *DATA* пина обезбеђује двострану комуникацију и користи се за трансфер података од и до сензора. Приликом слања података до сензора *DATA* пин користи узлазну ивицу серијског такта. Вредност *DATA* пина мора остати константна све док је такт високо.

Вредност података се може изменити са силазном ивицом такта. Како би се избегло преслушавање у сигналима, микроконтролер врши пренос података ниским напоном. Како би се постигао високи напон на линији за податке, уводи се додатни отпорник отпорности 10 kΩ. Овакав отпорник се може повезати са улазно излазним колом микроконтролера. Дати отпорник на слици 16 стављен је у везу са линијом за податке и линијом напајања. Иницијализација *SHT15* сензора након постављања и повезивања на линују напајања врши се у трајању од 11 ms. Након истека овог времена сензор прелази у мод мировања. Након одласка у мод мировања може се покренути трансмисија слањем трансмисионе старт секвенце. За овај тип сензора карактеристично је адрсирање помоћу 8-битова, при чему се подсеквентна команда састоји од три адресна бита и пет командних битова. Адресни битови се увек представљају нулама, док је листа командних битова дата у табели 3. На основу датих адресних и командних битова креирају се наредбе на основу којих микроконтролер комуницира са датим сензором.

Табела 3: SHT1x листа команди

Опис команде	Kod komande
Резервисано	000x
Мерење температуре ваздуха	00011
Мерење релативне влажности ваздуха	00101
Читање статус регистра	00111
Упис у статус регистар	00110
Резервисано	0101x-1110x
Ресет	11110

На пример, наредбе за мерење релативне влажности ваздуха и темпераутуре ваздуха изгледале би 00000101 и 00000011 респективно. Време одзива овог сензора разликује се у зависности од тога да ли се ради о мерењу на 8, 12 или 14 битова, и износи 20 ms, 80 ms, 320 ms респективно. Команда ресета, у облику наведеном у табели 3, обавља функцију ресетовања интерфејса сензора, и постављање статуса регистра на подразумеване вредности. Подразумеване вредности резолуције статуст регистра износе 14-битова за температуру и 12-битова за релативну влажност ваздуха.

Подразумевани број битова за резолуцију релативне влажности ваздуха и температуре ваздуха може се променити на 12 и 8 битова. Након ресетовања време чекања до наредне комаде износи 11 ms. Очитане вредности релативне влажности и температуре морају се израчунати како би се исте могле користити у даљим корацима истраживања. С обзиром на чињеницу да сензор релативне влажности нема линеарне карактеристике, потребно је извршити конверзију очитаних вредности. Уколико се очитане вредности означе са SO_{RH} на основу једначине дате испод, као и вредности коефицијената датих у табели 4, може се израчунати вредност релативне влажности ваздуха.

$$RH_{linear} = c_1 + c_2 \cdot SO_{RH} + c_3 \cdot SO_{RH}^2 \quad [\%RH]$$

Табела 4: Коефицијенти за конверзију релативне влажности ваздуха

SO_{RH}	C_1	C_2	C_3
12 битова	-2,0468	0,0367	-1,5955E-6
8 битова	-2,0468	0,5872	-4,0845E-4

На основу дате једначине све вредности релативне влажности ваздуха које износе преко 99% сматрају се потпуно засићеном ваздухом и могу се представити као 100% RH. Температурни сензор, за разлику од сензора за релативну влажност ваздуха, одликује се линеарношћу. Вредност температуре се добија израчунавањем уз примену корекционих коефициената датих у табели 5. Уколико се измерена вредност температуре представи са SO_T стварна вредност температуре добија се на основу дате једначине:

$$T = d_1 + d_2 \cdot SO_T$$

Посматрањем табеле са корекционим коефицијентима, може се закључити да вредност корекционих коефицијената зависи од вредности напона сензора. Такође, потребно је нагласити да се вредности ових коефицијената разликују уколико се температура приказује у фаренхајтима. Из практичних разлога, због потреба будуће имплементације у табели 5 су дати коефицијенти само за степене целзијусове, с обзиром да је ово подразумевана јединица мере температуре на подручју на коме је рађено истраживање.

Табела 5: Коефицијенти за конверзију температуре ваздуха

VDD [V]	d_1 [ºC]	SO_T	d_2 [ºC]
5,0	-40,1	14 bit	0,01
4,0	-39,8	12 bit	0,04
3,5	-39,7		
3,0	-39,6		
2,5	-39,4		

Додатна погодност приликом креирања метеоролошке станице употребом оваквог сензора јесте и израчунавање тачке росе. Као што је био случај са вредностима релативне влажности ваздуха и темепературе ваздуха, вредност тачке росе се не може добити мерењем, већ се мора израчунати. Како се вредности релативне влажности ваздуха и температуре ваздуха добијају мерењем на истом чипу, SHT15 омогућава овакво израчунавање. Уколико се тачка росе израчунава за измерени температурни опсег од -40 ºC до 50 ºC, може се применити следећа једначина.

$$T_d(RH,T) = T_n \cdot \frac{\ln\left(\frac{RH}{100\%}\right) + \frac{m \cdot T}{T_n + T}}{m - \ln\left(\frac{RH}{100\%}\right) - \frac{m \cdot T}{T_n + T}}$$

Вредности параметара T_n и параметра m које се користе приликом израчунавања тачке росе дате су табели 6.

Табела 6: Параметри за израчунавање тачке росе

Температурни опсег	Tn [ºC]	m
Вредности изнад нуле 0 ºC – 50 ºC	243,12	17,62
Вредности испод нуле -40 ºC – 0 ºC	272,62	22,46

У датој једначини *ln* представља природни логаритам. Вредности релативне влажности ваздуха представљене са *RH* и температуре ваздуха представљене са *T* због нелинеарности мерења требају бити вредности добијене израчунавањем према напред наведеним једначинама и корекционим коефициентима.

Према свим наведеним карактеристикама одабрани сензор за мерење релативне влажности ваздуха и температуре ваздуха заједно са могућношћу израчунавања тачке росе задовољава све задате критеријуме у циљу реализације потребних мерења.

5.2.3 Сензор за мерење количине падавина

Метеоролошке станице намењене прикупљању података за потребе пољопривредне производње морају бити опремљене уређајем за мерење дневне количине падавина. У највећем броју случајева, дневна количина падавина се мери у милиметрима воденог талога на једном квадратном метру површине током једног дана. Из ових разлога, уређај за мерење количине падавина је саставни део пројектованог модела метеоролошке станице. Сви уређаји за мерење количине падавина доступни на тржишту раде по истом принципу, те није било потребе за неким детаљнијим анализама и поређењима. Одабрани уређај за мерење количине падавина у основи је посуда која се сама празни, што је приказано на слици 17.

Слика 17: Изглед унутрашњости уређаја за мерење количине падавина, слика преузета са: https://projects.raspberrypi.org/en/projects/build-your-own-weather-station/9

Овај уређај је направљен као левкасти суд у коме се сакупља кишница и каналише до посуда за сакупљање. Када се сакупи довољна количина воде, долази до превртања посуде и њеног пражњења. У исто време, посуда на супротној страни се позиционира у положај за пуњење. Количина воде која је потребна да би извршила превртање посуде и њено пражњење износи 0,2794 mm кишнице. Дати податак показује да је уређај пројектован тако да евидентира и мале количине падавина.

Са сваким пражњењем посуде врши се и затварање електричног кола помоћу прекидача, што се евидентира помоћу дигиталног бројача или прекидом микроконтролера. Практично свако пражњење узрокује инкрементирање бројача. Укупна количина падавина за дати период израчунава се множењем вредности бројача и количине воде потребне за иницирање пражњења. Унутар гребена између поменуте две посуде налаизи се мали цилиндрични магнет који показује према задњем зиду. У самом задњем зиду налази се поменути прекидач. Прекидач поседује два метална контакта унутар њега који ће се додирнути када су под утицајем магнета. Према томе, посматрано са електронске стране, овај механизам функционише на исти начин као и било које дугме које је повезано са микроконтролером. Практично, када се једна од посуда преврне, магнет пролази прекидач и узрокује његово тренутно затварање. Уколико је прекидач повезан са *GPIO* пионом на микроконтролеру његово затарање ће генерисати ниски сигнал који се може открити и евидентирати као инкремент за један. Овакав уређај за мерење количине падавина може се повезати на два начина. Уобичајено долази опремљен *RJ11* прикључком иако користи само две жице, једну црвену и једну зелену. Уколико на микроконтролеру постоји могућност повезивања преко RJ11 утикача, исти се може искористити. Са друге стране, уколико оваква могућност не постоји, може се повезати преко улазно излазних пинова опште намене уклањањем утикача и повезивањем помоћи жица. Предност овог уређаја је рад без спољашњег напајања. Рад уређаја је заснован на механичким принципима, те не захтева било спољашње напајање, било напајање преко микроконтролера.

5.2.4 Сензор за мерење брзине и смера ветра

Брзина и смер ветра, такође, су два значајна параметра метеоролошке станице. С тим у вези, развијени модел метеоролошке станице опремљен је и сензорима за мерење ових параметара. Сензор за мерење брзине ветра се традиционално назива анемометар и састоји се од неколико полулопти које под утицајем ветра ротирају на заједничкој осовини. У највећем броју случајева најпростији механизам рада се заснива на механичком деловању магнета на прекидач, као и код сензора за количину падавина. У табели 7 дато је поређење различитих анемометра доступних на тржишту.

Табела 7: Поређење карактеристика различитих сензора за мерење брзине и смера ветра

Ознака	Стартна брзина [m/s]	Мерни опсег [m/s]	Тачност [m/s]	Напајање [V]	Излазни сигнал [V]	Цена [din]
Adafruit1733	0,2	0,5 - 70	± 1	7 - 24	0,4 - 2	5500,00
BAN 1093558	0,2 - 0,4	0,2 - 32,4	± 1	7 - 24	0,4 - 2	6200,00
JL-FS2	0,4 - 0,8	0,0 - 30	± 1	9 - 24	0 - 5	8000,00
Met One 010C	0,22	0,0 - 50	± 0,07	12	11	29000,00
Met One 013	0,45	0,0 - 67	± 0,11	12	11	20000,00

Као што се може видети из дате табеле, основна разлика између наведених анемометра заснива се на прецизности мерења, потребном напајању и на осетљивости самог анемометра. Под осетљивошћу анемометра сматра се минимална јачина ветра која је потребна да покрене ротацију и самим тим иницира мерење. Одабрани анемометар приказан је на слици 18. Значајно за одабрани анемометар, с обзиром да је намењен примени на отвореном, јесте да је отпоран је на утицај временских услова, корозију и утицај влаге.

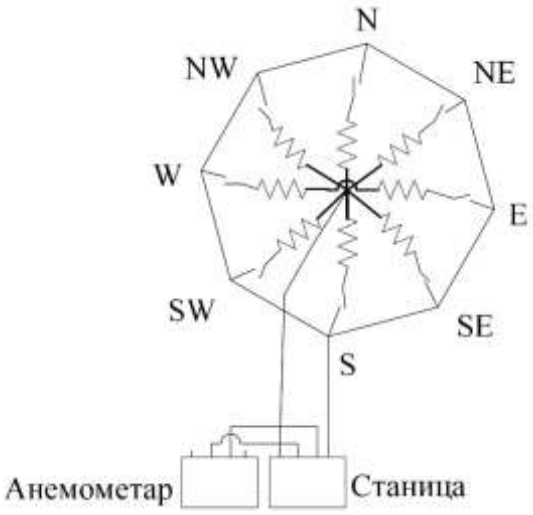
Слика 18: Изглед одабраног анемометра, слика преузета са: https://www.adafruit.com/product/1733

Материјали од којих је направљен обезбеђују трајност уређаја и тачност приликом мерења брзине ветра. Тачност овог анемометра, као што се може видети, веома је задовољавајућа и износи ±1 m/s. Брзина ветра потребна за покретање мерења код овог типа анемометра обезбеђује мерење ветра готово најмање јачине. Максимална брзина ветра коју дати анемометар може евидентирати далеко је виша од брзине ветра потребне за истраживања у области предикције и износи 70 m/s.

Са друге стране, горња мерна граница брзине ветра пружа својеврстан вид заштите уређаја од могућих јаких ветрова који би уређаје са мањим прагом мерења евентуално оштетитили. Исто тако, за поменути сензор дефинисано је и ограничење у виду максималне спољашње температуре на којој анемометар може радити и која износи од -40ºC до +80ºC. Притом, овај анемометар је отпоран на директно сунчево зрачење, што значи да утицај *UV* зрачења неће утицати на материјале од којих је анемометар направљен, те самим тим и на квалитет мерења. Одабрани анемометар долази са каблом за напајање и пренос очитаних вредности дужине 3 m, што омогућава његово позиционирање на врх носача, што је приказано на слици 4.

Повезивање анемометра обезбеђено је преко три жице од којих браон жица представља напајање и одговара пину 1, црна жица представља уземљење и одовара пуну 2, док последња плава жица одговара пину 4 и не повезује се. Како потребно напајање превазилази могућности напајања са микроконтролера (у случају овог модела метеоролошке станице *Raspberry Pi* микроконтролера), потребно је анемометар повезати на спољашње напајање. Овакав вид реализације напајања са једне стране уводи потребу за коришћењем додатног хардвера како би се са батерије или сталног напајања довело напајање до анемометра, док са друге стране оставља могућности напајања других сензора са микроконтролера.

Показивач смера ветра је још један од сензора који спадају у групу метеоролошких сензора. Посматрано из домена функционалности, може се сматрати да је овај сензор можда и најсложенији од свих метеоролошких сензора. Уколико се посматра механичко решење овог сензора, при чему би исти радио без додатног улазног напајања, показивач смера ветра би се састојао од осам преклопника од којих је сваки повезан са различитим отпорником. Шематски приказ рада показивача правца ветра дат је на слици 19. Сваки од осам отпорника има различиту отпорност, што обезбеђује шеснаест различитих могућих комбинација отпорности. Магнет у оквиру показивача правца ветра приликом рада може затворити два преклопника одједном што омогућава показивање до 16 различитих позиција. Ово се дешава у моментима када је позициониран на пола пута између два преклопника.



Слика 19: Шема показивача правца ветра

Како би се очитао правац ветра са показивача правца ветра потребно је измерену отпорност добијену са сензора конвертовати у аналогну вредност. Уместо директног мерења отпорности, једноставнији приступ је мерење напона на показивачу правца који варира у односу на то која комбинација отпорника је тренутно укључена у коло помоћу преклопника, што значи да се врши мерење аналогне вредности. Показивач правца ће константно као излазну вредност давати опсег напона. На овакав начин се креира излазни напон који се може измерити помоћу *AD* конвертора. Потреба за коришћењем додатног *AD* конвертора произилази из чињенице да *Raspberry Pi* поседује само дигиталне улазе. Један од популарнијих *AD* конвертора је *MCP3008*. Интегрисано коло овог конвертора састоји се од 16 пинова са 8 аналогних улаза који се могу користити као коло за конверзију. Такође, поседује 10-битни *ADC*, односно има $2^{10} = 1024$ могућих излазних вредности за референтни напон од 5 V. Најмања вредност промене напона коју овај конвертер може детектовати износи 5 V/1024=4.88 mV.

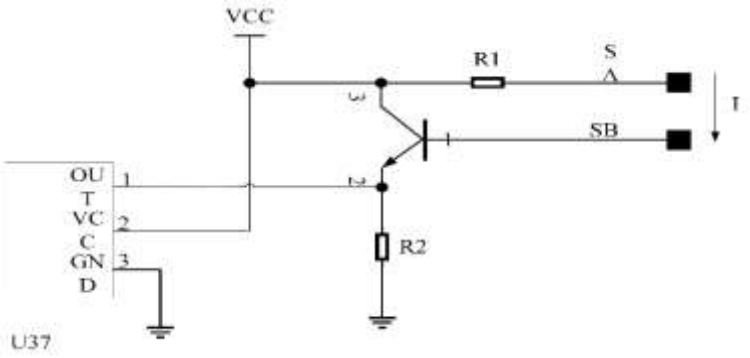
5.2.5 Сензор за мерење влажности земљишта

Улога сензора за мерење влажности земљишта огледа се у константном мерењу количне воде у земљишту у циљу одржавања одговарајуће влажности потребне за правилан узгој биљака. Сензор за мерење влажности земљишта се поставља испод нивоа земље на унапред дефинисаној дубини. Дубина постављања сензора зависи од биљне врсте која се гаји на датој површини.

Принцип рада сензора за мерење влажности земљишта доступних на тржишту заснива се на две сонде као што је приказано на слици 20. Кроз сонде се у земљу преноси електрична енергија, након чега се на основу измерене отпорности израчунава ниво влаге.

Слика 20: Octopus сензор влажности земљишта, слика преузета са: https://www.elecfreaks.com/store/octopus-soil-moisture-sensor-brick.html

Већа влажност земљишта омогућава бољу проводност, а самим тим и мању отпорност, док сува земља слабо проводи електричну енергију због већег отпора. Сензори за мерење влажности земљишта који су доступни на тржишту истоветних су карактеристика. Пресудни параметар у одабиру одговарајућег је била цена. За потребе реализације модела метеоролошке станице одабран је *Octopus* сензор за мерење влажности земљичта. Шематски приказ кола *Octopus* сензора дат је на слици 21.



Слика 21: Шема Octopus сензора

Овај сензор се напаја константним напајањем од 3,3 V до 5 V, што одговара потребама реализације метеоролошке станице, с обзиром на чињеницу да се напајање може врштити директно са *Raspberry Pi* уређаја или са спољашњег напајања. Излазни сигнал којим се преноси измерена вредност је у границама од 0 – 4,2 V, што одговара потребама реализације модела. Уколико се посматра распоред пинова, пин 1 представља аналогни излаз и повезује се жутом жицом, пин 2 одговара напајању и повезујесе црвеном жицом, док пин 3 одговара уземљењу и повезује се црном жицом.

5.2.6 Сензор за мерење влажности листа

Влажност листа биљке или присуство воде на листу биљке може се сматрати једним од веома значајних параметара у процесу предикције појаве болести код већине гајених биљака. Појава влажности листа биљке изучава се деценијама уназад, посебно у области фитопатологије и пољопрвиредне метеорологије. Директно мерење влажности листа је веома проблематично, јер је веома тешко повезати сензор на лист биљке. Сама позиција листа, његова изложеност сунцу и здравствено стање стално су променљиви. Како би се избегли ови проблеми, развијени су сензори који врше процену могуће влажности листова у чијој се близини налазе. Принцип рада ових сензора заснива се на мерењу диалектричне константе горње површине сензора. Овакав сензор може детектовати присустно најмање количине воде или леда на својој површини. Како се на тржишту могу наћи различити сензори за мерење влажности листа у табели 8 су приказани сензори обухваћени анализом.

Табела 8: Поређење карактеристика сензора за мерење влажности листа

Ознака	Напајање [V]	Излазни напон [mV]	Радна температура [ºC]	Дужина кабла [m]
Decagon	2,5 – 5,0	320 - 1000	-20 do +60	5
PHYTOS 31	2,5 – 5,0	320 – 1250	-40 do +50	5
Vantage Pro	3,0	2500 – 3000	-20 do +60	5
ADCON Wet	2,2 – 12,0	0 - 2500	-20 do +60	3
260-RK300-04	12,0 – 24,0	0 - 5000	-40 do +70	3

Из групе анализираних сензора приказаних у табели 7, на основу потребних карактеристика, одабран је *Decagon* сензор. Изглед овог сензора дат је на слици 22. Принцип рада овог сензора се заснива на детекцији промене електричне отпорности између позлаћених елемената мреже од које је направљен. Као и остали сензори ове групе *Decagon* сензор мери диелектричну константу у зони приближно 1 cm изнад површине сензора. Познато је да је диелектрична константа воде 80, а леда 5, што је знатно веће од диелектричне константе ваздуха која износи 1.

Слика 22: Изглед и шема Decagon сензора, слика преузета са:
http://www.ictinternational.com/products/lws/phytos-31-leaf-wetness-sensor/

Сензор емитује сигнал у mV који је пропорционалан диелектричној вредности у зони мерења и самим тим пропорционалан количини воде или леда на површини сензора. Дебљина фибергласа од кога је направљен овај сензор износи 0,65 mm, што је приближно дебљини листа биљке. Уколико је топлота листа процењена на 1425 J m^{-2} K^{-1}, а његова дебљина на 0,4 mm онда је капацитет топлоте сензора, с обзиром на његову дебљину, 1480J $m^{-2}K^{-1}$. Управо из овог разлога, кондензација влаге на сензору и њено испарење се одиграва истом брзином као на нормалном лишћу. Временски период једног очитавања износи 10 ms. Овај сензор може се монтирати на самој метеоролошкој станици или у оквиру биљке међу њеним лишћем. Уколико се сензор монтира на биљци, дужина кабла се може повећати у односу на иницијалних 5 m без утицаја на само мерење и пренос очитаних вредности. Процес повезивања сензора може се реализовати на два начина. Први од начина је коришћење *Decagon data logera*, док је други од

начина коришћење неког другог уређаја у замену за data loger. Како се у случају предложеног модела метеоролошке станице не користи Decagon data loger, потребно је прекинути оригинални прикључак и повезивање урадити помоћу жица које се налазе у самом каблу. Притом, бела жица се користи за напајања, *bare shield* за уземљење и црвена жица за пренос сигнала.

У случају коришћења другог *data logera* потребно је програмирати *loger* тако да добијене измерне вредности распореди у правилном облику. Када се ради о *Raspberry Pi* уређају, подаци се могу сместити у *Excel* фајл. Такође, у случају било ког другог *data logera* морају се одредити границе влажности листа. Ово значи да излазне вредности у случају сувог листа, као и у случају потпуно мокрог листа, зависе од вредности напона који се доводи на дати сензор. Уобичајено је да се за вредност влажног листа узима вредност мало већа од вредности сувог листа. У оваквом случају очитане вредности се упоређују са вредностима сувог сензора како би се одредио ниво влажности. Како би се тачно одредила влажност листа, као и трајање влажности листа потребно је радити очитавања у временском размаку од 15 или мање минута. Степен влажности листа и трајање влажности листа подједнако су значајни параметри у прогнози остварености услова за појаву биљних болести. С обзиром на сталну изложеност сензора прашини и утицају спољашњих услова, потребно је његово периодично чишћење у циљу детекције тачних вредности влажности листа.

5.3 Компоненте за прикупљање просторно-временских параметара

Просторно временска компонента сваког од пакета података послатих са меторолошке станице до базне станице, као што је раније наведено, значајна је у домену постојања система метеоролошких станица повезаних на једну базну станицу. Притом ове две комоненте се могу раздвојити на просторну и временску компоненту. Без обзира да ли постоји једна или више метеоролошких станица, временска компонента је значајна због евидентирања времена очитавања параметра, као и праћења промена метеорлошких вредности током периода трајања дана. Просторна компонента, са друге стране, није потребна уколико постоји само једна метеоролошка станица и уколико је она увек на истој локацији.

Просторно-временска компонента је укључена у оквиру модела метеоролошке станце управо због постојања потребе за реализацијом система који садржи више од једне метеоролошке станице. За потребе одређивања просторно-временских координата у модел метеоролошке станице је уведен GPS модул. Због постојања већег броја хардверских решења GPS модула која се могу искористити и повезати са *Raspberry Pi* рачунаром, извршена је анализа у циљу одабира наједакватнијег. Карактеристике поређених модела дате су у табели 9.

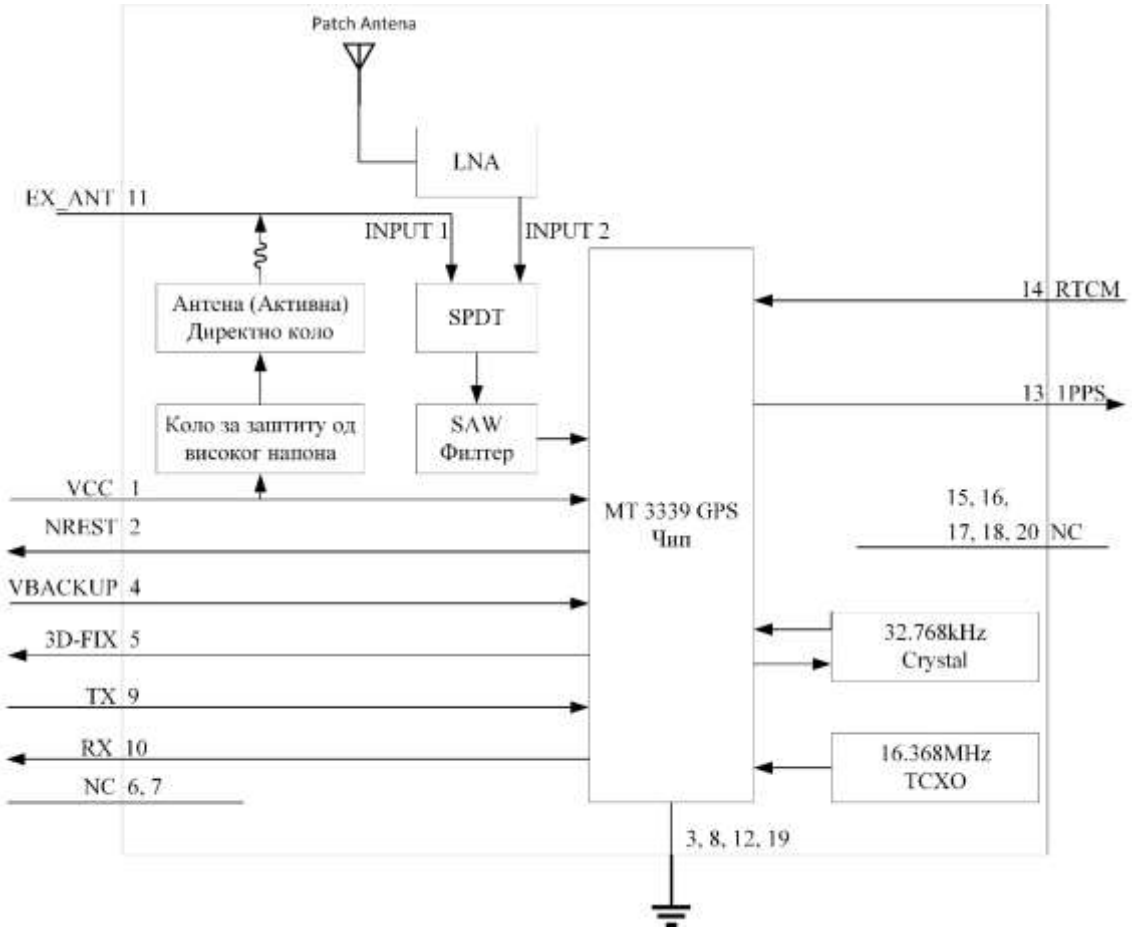
Табела 9: Анализа GPS модула

Ознака	Тачност [m]	Напајање [V]	Време стартовања [s]	Опсег освежавања [Hz]	Baud rate [bps]	Цена [din]
Adafruit PA6H1F1702	3	3,0 - 5,5	34	1 - 10	9600	5000,00
MediaTek GPS-2261	3	5,0	36	1 - 10	9600	3000,00
GPS-NEO-6M-001	3	3,0 - 5,0	36	1 - 10	38400	2000,00
PA6C1V1438	3	5,00	36	1-10	9600	1800,00

Поред основних карактеристика GPS модула, посебан акценат стављен је на повезивање истог са *Raspberry Pi* рачунаром. Наиме, постоји велики број GPS модула који према перформансама одговарају потребама метеоролошке станице, који су, међутим, иницајлно намењени и компатибилни са другим микрокотролерима. У оваквом случају њихово повезивање са *Raspberry Pi* рачунаром могуће је уз увођење додатне хардверске плоче која би се користила као веза између датог модула и *Raspbberry Pi* рачунара. На овакав начин би се извршило заузеће свих *GPIO* пинова, па самим тим повезивање других, већ поменутих сензора, не би било могуће. Управо из ових разлога, одабир одговарајућег GPS модула оријентисан је у смеру могућег повезивања помоћу *USB* порта. Одабрани *Adafruit* GPS модул је приказан на слици 23. Хардверско решење овог модула је изграђено око *MTK3339 chipset*-а, који представља GPS модул високог квалитета, који може пратити до 22 сателита кроз 66 канала. Овај GPS модул има високо осетљиви пријемник од -165 dBm приликом праћења, као и уграђену антену.

Слика 23: Изглед одабраног Adafruit GPS модула, слика преузета са: https://www.adafruit.com/product/746

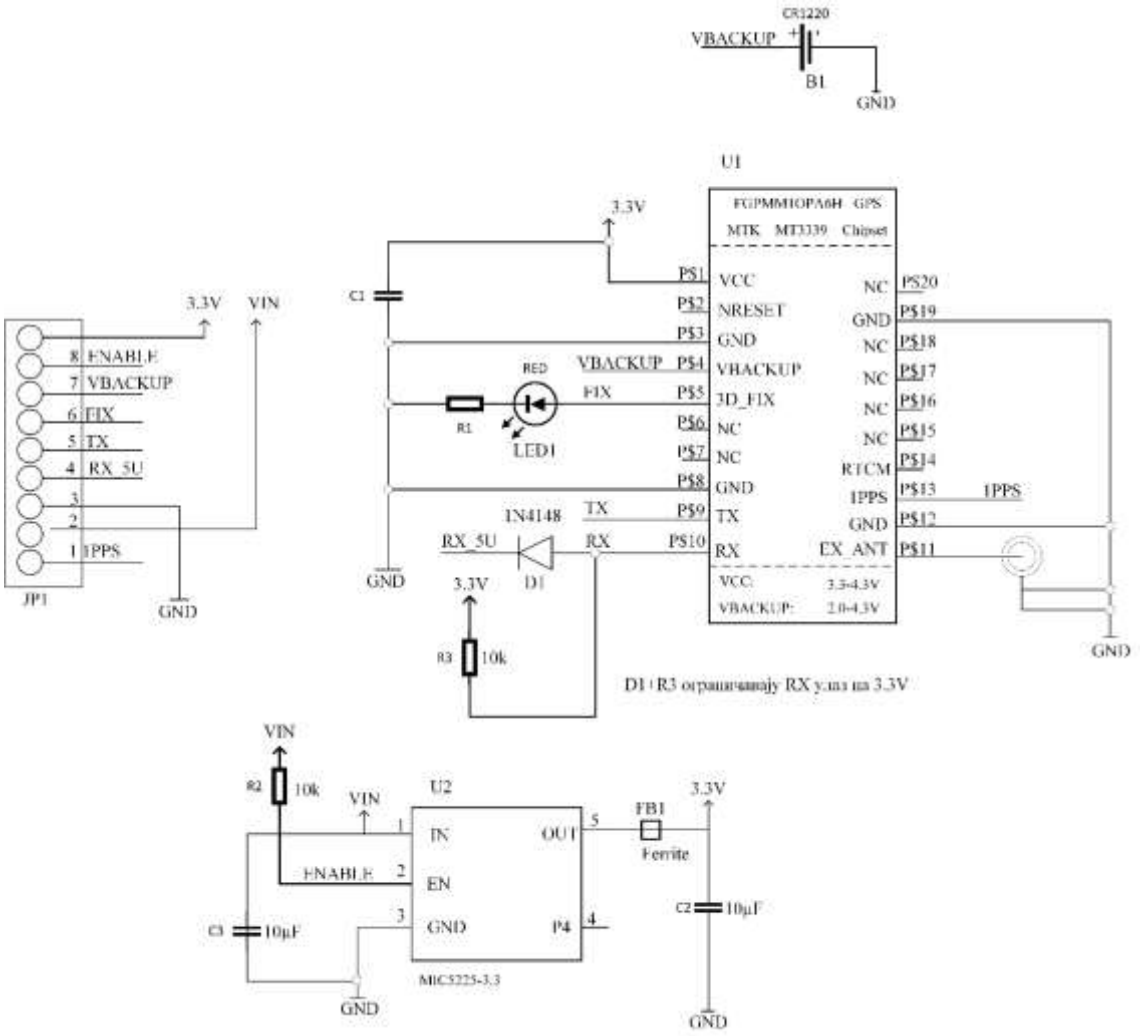
Одликује се малом потрошњом енергије до 20 mA приликом одређивања локације, као и модом мировања који се одликује још мањом употребом енергије. У верзији 3 овог модула могуће је повезивање спољашње антене која омогућава прецизније одређивање позиције. Повезану антену модул сам открива и прелази на коришћење ово антене без додатних интервенција корисника. Додатна погодност овог модула је уграђени *data loger*. Практично, унутар модула се налази микроконтролер са празном *flash* меморијом који омогућава очитавање и памћење података без сталне интервенције *Rasberry Pi* рачунара. Задатак *Raspberry Pi* рачунара сведен је на издавање команде „*Start Logging*“, након чега овај рачунар може прећи у мод мировања наредних 16 сати. Током ових 16 сати тачно време, датум, географска ширина, дужина и надморска висина очитавају се и памте у *flesh* меморију на сваких 15 секунди. Очитани подаци се аутоматски надовезују, тако да не постоји могућност губитка података услед нестанка напајања или неког другог хазарда. Дефинисану фреквенцу очитавања и параметре који се региструју није могуће изменити јер су од стане произвођача хардверски дефинисани унутар модула. На овакав начин се врши уштеда енергије коју би *Rasberry Pi* рачунар морао да троши позивајући очитавање просторно-времснких података током поменутих 16 часова. Блок дијаграм *MTK3339* модула је приказан на слици 24, док је шема повезивања овог модула са *Adafruit* плочом дата на слици 25.



Слика 24: Блок дијаграм MTK3339 GPS модула

У случају повезивања екстерне антене, пожељно је да се постави тако да има слободан простор према небу како би се добили најбољи резулатати. Са датог блок дијаграма може се видети да и уграђена и екстерна антена доводе улазни сигнал у GPS чип. За успешан рад екстерне антене потребно је обезбедити напајање у опсегу од 3 V до 3,6 V. У циљу одређивања активне антене и одабира између коришћења уграђене или екстерне антене користи се софтвер дизајниран за детекцију активне антене.

Повезивање путем *USB* порта је одабрано због тога што су ови портови на *Rasberry Pi* уређају слободни и неискоришћени. Реализација оваквог начина повезивања захтева увођење додатног хардвера у виду адаптера са *USB* на *TTL*. На слици 26 приказан је одговарајући адаптер. Цена увођења оваквог вида додатног хардвера је занемарљива у поређењу са добијеним перформансама.



Слика 25: Шема комплетног GPS модула верзије 3

Унутар *USB* конектора налази се чип који обезбеђује конверзију између серијске везе у *USB* и обратно. На крају кабла се налазе четири жице којима се врши повезивање. Црвена жица се користи за напајање, црна жица се повезује са уземљењем, зелена жица се повезује *USB TX* са *RX* на GPS модулу, док бела жица повезује *RX* на *USB*-у са *TX* на GPS-у. Пин напајања обезбеђује 5 V или 500 mA напајања директно са *USB* порта, док *RX/TX* пинови раде на 3,3 V и обезбеђују повезивање са већином логичких *chipset*-ова. Једном када се овако повезан адаптер прикључи на USB порт Raspberry Pi рачунара, биће препознат као *dev/tty/USB0*, уколико не постоји повезан ни један други *ttyUSB* адаптер. Уколико пак постоји, нула ће бити замењена другим бројем. Како би се обезбедила потребна комуникација између *Raspberry Pi* рачунара и GPS модула који прослеђује серијске податке, потребно је извршити инсталацију неког од софтвера

који ће разумети серијске податке који са *GPS* модула до рачунара долазе преко */dev/tty/USB0* адаптера. Један од могућих софтвера је и *GPS Deamon (gpsd)* који је садржан у самом хардверу и који се понаша као слој између апликација и стварног *GPS* хардвера и врши обраду грешака и обезбеђење интерфејса за било који *GPS* модул.

Слика 26: Изглед USB на TTL адаптера, слика преузета са: https://www.adafruit.com/product/954

Други начин била би инсталација *driver*-а за поменути адаптер. У случају коришћења *Linux* оперативног система, драјвери су већ укључени у оквиру kernela, тако да њихова инсталација није потребна, док је у случају *Windows* или *Mac* окружења њихова инсталација потребна. У случају *Window* окружења на коме је заснован рад *Raspberry Pi* уређаја треће генерације могуће је инсталирати два *driver*-а, у зависности од тога који *chipset* кабла је искоришћен приликом харверске реализације. За старије каблове се користи *Prolific* бренд *driver*-а, док се за новије верзије каблова, настале након 2017. године, користи *SiLabs* бренд *driver*-а. Инсталација једног типа *driver*-а не искључује инсалацију другог.

5.4 Напајање метеоролошке станице

Посматрано са становишта напајања моделом, описана метеоролошка станица представља скуп електронских уређаја за чији је неометани рад потребно обезбедити константан извор напајања током целог периода коришћења. Како је моделом описана метеоролошка станица намењена коришћењу на производним пољопривредним површинама, основна претпоставка је да не постоји дистрибутивна електро мрежа која би се искористила као извор напајања.

Претпоставка непостојања извора напајања путем дистрибутивне електро мреже оријентише реализацију напајања у смеру проналажења алтернативних могућности за напајање метеоролошле станице. Са друге стране, потенцијално постојање дистрибутивне електро мреже у подручју на коме би метеролошка станица била лоцирана, као и њено коришћење за потребе напајања метеорлошке станице, елиминисало би потребу за уградњом алтернативних хардверских решења. У исто време, постојање дистрибутивне електро мреже довело би и до ниже цене реализације метеоролошке станице. Моделом метеоролошке станице датим на слици 4 предвиђено је увођење алтернативног вида напајања помоћу фотонапонског панела и акумулаторске батерије. Реализација напајања метеоролошке станице на овакав начин треба да обезбеди њено активно коришћење током целе године. Увођење акумулаторске батерије обезбеђује коришћење метеоролошке станице током дана у којима је редукован број сунчаних сати. У исто време, акумулаторска батерија у комбинацији са додатним хардверским компонентама за регулацију напона представља својеврсну заштиту микроконтролера, метеоролошких сензора, GPS модула и модула за слање података.

Raspberry Pi рачунар као једна од главних компоненти метеоролошке станице захтева константан напон од 5 V. Поред њега, сви одабрани сензори и уређаји, као што је раније наведено, раде на истом или мањем напону који добијају повезивањем на улазно излазне пинове или преко *USB* порта *Raspberry Pi* рачунара. Посматрано са становишта потрошње, за сваки од сензора и уређаја повезаних на *Raspberry Pi* рачунар могу се дефинисати два режима рада: активан и пасиван. Како би се одредила потребна снага фотонапонског панела и капацитет батерија, извршено је израчунавање потрошње целокупног система. Било је потребно до детаља сагледати све режиме рада уређаја који се напајају и теоријски најнеповољнији режим рада система који се напаја. Најнеповољнији услов значи да је потребно сагледати активан режим рада потрошача када он троши највише електричне енергије. У табели 10 је дат преглед потрошње електричне енергије за сваког од потрошача дефинисаних у оквиру мреже. С обзиром да је већа потрошња енергије у активном режиму рада, ова вредност је узета као референтна за даље израчунавање снаге фотонапонског панела и

капацитета батерије. Како би се израчунала потрошња струје у часу главног оптерећења, дефинисана су одређена правила рада метеоролошке станице. Према овим правилима, очитавање вредности метеоролошких параметара се врши четири пута у току једног сата, односно на сваких 15 минута.

Табела 10: Појединачна потрошња свих уређаја у систему метеоролошке станице

Уређај	Режим рада активан/пасиван	Јачина струје при 5V DC [mA]	Трајање режима рада [sec]	Потрошња струје [mAh] за 1h
Raspberry Pi	*активан*	*2500*	*3600*	*2500*
DS18B20	пасиван	0	3360	0
	активан	*1,5*	*240*	*0,1*
SHT15	пасиван	0,001	3360	0,00093
	активан	*0,6*	*240*	*0.04*
Adafruit1733	пасиван	24	3360	22,4
	активан	*40*	*240*	*2,67*
Octopus	пасиван	0	3360	0
	активан	*35*	*240*	*2,33*
Decagon	пасиван	2	3360	1,87
	активан	*7*	*240*	*0,47*
Adafruit PA6H1F1702	пасиван	0	3360	0
	активан	*20*	*240*	*1,33*
SIM 900 GSM modem	пасиван	1	3360	0,93
	активан	*590*	*240*	*39,33*

Дефинисано максимално време које сензор приликом очитавања проведе у активном режиму рада износи један минут, што је више од времена предвиђеног техничком документацијом. Разлика у трајању између активног режима рада предвиђеног техничком документацијом за сваки од сензора и дефинисаног времена од једног минута, уведена је у прорачун како би се покриле ситуације појаве грешке приликом очитавања, што узрокује покретање процеса поновног очитавања. С обзиром да се ради о ведностима потрошње у mAh, повећање времена рада не утиче значајно на даљи прорачун снаге фотонапонског панела и капацитет акумулаторске батерије. Такође, посматрајући табелу 10 може се уочити да сензор за мерење количине падавина није наведен.

Због механичке природе рада сензора, овај сензор не захтева утрошак енергије. Последње наведени потрошач, који је детаљније описан у наредном поглављу, представља GSM модул који је одабран за потребе преноса измерених вредности параметара од метеоролошке станице до базне станице. Овај модем се повезује и са *Raspberry Pi* рачунаром, те је и он морао бити обухваћен прорачуном потрошње. На основу података дефинисаних у табели 10 следи да току једног часа потрошња метеоролошке станице у пасивном могу рада износи 25,2 mAh, док у активном режиму рада потрошња износи 2546,273 mAh. Како је циљ сагледати укупну потрошњу свих уређаја метеоролошке станице у току 24 h и то у најнеповољнијем режиму рада усвојено је да метеоролошка станица ради у активном режиму рада. На основу веће потрошње струје у активном режиму рада, укупна дневна потрошња струје свих уређаја у оквиру моделом дефинисане метеоролошке станице износи $I_0 = 61{,}11056\text{Ah}$. Напајање свих потрошача врши се једносмерним напоном $U_0 = 5\text{V}$, при чему се долази до податка о дневној потрошњи електричне енергије која износи:

$$P_0 = U_0 \cdot I_0 = 5V \cdot 61{,}11056Ah = 305{,}5528\text{Wh}$$

На основу израчунате дневне потрошње електричне енергије може се израчунати снага фотонапонског панела која представља снагу електричне енергије коју панел генерише при интензитету сунчевог зрачења од 1000 W/m^2. Како би се израчунала потребна снага фотонапонског панела, најпре се мора ирачунати такозвани средњи број сати вршног сунца у најнеповољнијем месецу. Ово практично значи да се врши нормализација снаге сунчевог зрачења у односу на референтну вредност од 1000 W/m^2. Математички посматрано средњи број сати вршног сунца рачуна се по формули:

$$N_{PS} = \frac{H(W_h/m^2)}{1000(W/m^2)}$$

При чему $H(W_h/m^2)$ представља просечан интензитет сунчевог зрачења у најнеповољнијем месецу током године. Како би се овај податак израчунао потребно је одредити азимут и елевацију фотонапонског панела. Најоптималније искоришћење фотонапонског панела се добија када се панел монтира на носач са

променљивим углом азимута и елевације. На овакав начин се може помоћу контролера вршити подешавање најоптималнијег угла оријентације панела. Пракса је да се у летњим месецима бира мањи угао елевације, док се у зимским месецима бира већи угао елевације. Конструктивно најпростија варијанта јесте да се панел монтира на фиксан носач без могућности промене елевације. У оваквом случају се израчунавање угла своди на одређивање угла елевације за најнеповољнији месец у години. Добијена вредност угла представља угао од хоризонталне равни. Одређивање месеца са најмањим интензитетом сунчевог зрачења и најоптималњег угла елевације панела за тај месец условљено је локацијом на којој ће панел и уређаји које ће он напајати бити постављен. За потребе овог истраживања одабрана локација на којој ће метеоролошка станица бити постављена налази се на следећим географским координатама 43.163791N, 21.362126E.

Израчунавање месеца са најмањим интензитетом сунчевог зрачења извршено је помоћу сервиса креираног од стране Центра за обједињена истраживања европске комисије (енг. *Europian Commission – Joint Research Center*)[2]. Након уноса географских координата и одабира опције под називом *Optimal incliation angle* добијен је резултат израчунавања на основу кога је за дату локацију месец децембар месец са најнижим интензитетом сунчевог зрачења. На основу генерисаног извештаја од стране овог сервиса закључује се да потребни елевациони угао фотонапонског панела износи 63º у односу на хоризонталну раван. Добијена вредност угла се у наставку користи за израчунавање просечне дневне суме глобалног зрачења по квадратном метру коју примају модули конкретног система. Вредност овог параметра, наког израчунавања од стране сервиса, износи 2,06 kWh/m^2. На основу овог податка се долази до битних параметара за даље израчунавање, односно до констатације да просечан интензитет сунчевог зрачења у децембру месецу износи $H = 2060\ Wh/m^2$. Заменом добијене вредности за интензитет сунчевог зрачења у оквиру једначине за број сати вршног сунца у најнеповољнијем месецу добија се следећа вредност:

[2] Europian Commission – Joint Research Center, доступно на: http://re.jrc.ec.europa.eu/pvgis/apps4/pvest.php, датум прегледа: 15.10.2017.

$$N_{PS} = \frac{H(Wh/m^2)}{1000(W/m^2)} = \frac{2060Wh/m^2}{1000(W/m^2)} = 2{,}06h$$

Снага фотонапонског панела *P(WP)*, који под овим условима треба да обезбеди напајање метеоролошке станице чија је дневна потрошња $P_0 = 305{,}5528\text{Wh}$, добија се на основу следећег израза:

$$P = \frac{P_0}{N_{PS} \cdot E_{REG} \cdot E_{BAT}}$$

При чему E_{REG} представља ефикасност регулатора пуњења, док E_{BAT} представља ефикасност батерија. За потребе израчунавања снаге фотонапонског панела вредности за ефикасност регулатора пуњења и ефикасност батерија узете су са $E_{REG} = 90\%$ и $E_{BAT} = 90\%$ респективно. На основу вредности свих параметара добија се снага фотонапонског панела:

$$P = \frac{P_0}{N_{PS} \cdot E_{REG} \cdot E_{BAT}} = \frac{305{,}5528Wh}{2{,}06h \cdot 0{,}9 \cdot 0{,}9} = 183{,}119W_P$$

На основу добијене вредности за снагу фотонапонског панела произилази да је потребно уградити два фотонапонска панела снаге од по 100 WP, јер се овакви панели могу наћи на тржишту.

Поред израчунавања снаге фотонапонског панела, извршено је и израчунавање капацитета батерије која је као што је већ наведено део система. Како би се израчунао потребан капацитет батерије, потребно је дефинисати услов расположивости система за напајање. Теоријски, услов расположивости система зависи од чињенице о томе колико је битно функционисање самог система који се напаја помоћу панела и акумулаторске батерије. У овом кораку прорачуна дефинише се колико времена батерија или батерије морају пружити аутономију рада потрошачима у условима такозваног тоталног мрака, односно у условима када панел не генерише никакву струју пуњења. Сходно потребама рада метеоролошке станице приликом прорачуна капацитета батерије узет је случај два узастопна дана тоталног мрака, у ознаци $N_{SOL} = 2$. Ово значи да је капацитет батерије довољан за два дана без енергије сунчевог зрачења или било каквог отказа на панелу.

Такође, што се тиче напона батерије за потребе прорачуна и реализације исте усвојен је напон батерије од $V_{BAT} = 12V$. Приликом прорачуна капацитета батерије, још један од битних параметра који је потребно узети у обзир јесте и ефикасност батерије. Посматрано са становишта ускладиштења енергије, због електро-хемијских процеса који се одвијају у батерији оне нису у стању да ускладиште сву електричну енергију коју добију, већ се део енергије троши и ослобађа у виду топлоте. За потребе прорачуна капацитета батерије усвојено је да фактор ефикасности износи $E_{BAT} = 80\%$.

Поред фактора ефикасности батерије потребно је дефинисати и такозвану дубину пражњења батерије. У зависности од технологије израде батерије, поставља се граница до које се батерија сме испразнити. Практично се не сме никада дозволити пражњење батерије до самог краја. За потребе регулисања дубине пражњења користи се регулатор пуњења чија је улога да покрене пуњење у тренутку када се батерија испразни до дозвољене дубине пражњења. У исто време, регулатор пуњења прекида довод струје ка потрошачима уколико се батерија испразни испод унапред дефинисане критичне границе. На овакав начин се врши очување батерије и њена дуготрајност. У случајевима када је батерија напуњена на 100% свог капацитета, фактор дубине пражњења батерије износи $D_{BAT} = 0\%$. Са пражњењем батерије расте и дубина пражњења. Сами тим, не постоји уопштена вредност дубине пражњења. Са већим пражњењем смањиће се радни век батерије, те ће у таквим случајевима бити потребна и батерија мањег капацитета, док ће са мањим пражњењем батерија дуже трајати, па ће у том случају бити потребна батерија већег капацитета. Генералне препоруке произвођача батерије по питању максималне дубине пражњења су да овај параметар треба да има вредност $D_{BAT} = 80\%$.

Последњи параметар који је потребно одредити како би се израчунао потребан капацитет батерије јесте деградација капацитета батерије услед пада температуре, у ознаци Ct_{BAT}. Капацитет батерије дефинисан од стране произвођача односи се на рад батерије у спољашњим условима при температури од 20ºC. Смањењем спољашње температуре смањује се и расположиви капацитет батерије.

Како би се одредио овај параметар потребно је познавати температуру окружења у коме ће систем радити заједно са батеријом. Тачније, потребно је познавати могућу најнижу температуру током зимског периода. Свакако, постоји могућност изолације батерије од спољашње температуре, што може ублажити утицај ниских температура током зимских месеци. У највећем броју случајева се усваја принцип огледан у томе да капацитет батерије опада за 1% са падом температуре за 1ºC испод 20ºC. С обзиром на чињеницу да је метеролошка станица пројектована и за рад у зимским месецима, усвојено је да батераја у таквим условима ради на температури од 0 ºC. Што значи да је њен расположиви капацитет услед пада температуре смањен на вредност $Ct_{BAT} = 80\%$.

Потребан капацитет батерије добија се заменом вредности свих наведених параметара у оквиру следеће једначине:

$$Q_{BAT} = \frac{P_0 \cdot N_{SOL}}{V_{BAT} \cdot E_{BAT} \cdot D_{BAT} \cdot Ct_{BAT}} = \frac{305.5528Wh \cdot 2}{12V \cdot 0{,}8 \cdot 0{,}8 \cdot 0{,}8} = \frac{611{,}1056Wh}{6{,}144V} = 99{,}46Ah$$

Када се ради о одабиру капацитета батерије, пракса је да се бира батерија која има први већи стандард у односу на израчунату вредност. За напајање моделом предложене метеорлошке станице потребна је батерија од 12 V капацитета 100 Ah.

Одабир батерије од 12 V и капацитета 100 Ah условљава увођење регулатора напона чији је задатак да снизи напон са 12 V на оперативни напон *Raspberry Pi* рачунара од 5 V. У исто време, регулатор напона је поред регулатора пуњења још једна хардверска компонета уведена ради очувања осталих хардверских уређаја од могућег оштећења услед промена напона.

Реализација моделом предложене метеоролошке станице са напајањем заснованим на коришћењу фотонапонског панела и акумулаторске батерије, у условима непостојања дистрибутивне електро мреже, представља једини вид стабилног напајања оваквог система. Посматрано са економског аспекта, креирање овакве метеоролошке станице је вишеструко исплативо. Ако се посматра само систем напајања, економске уштеде су вишеструке. Економске уштеде произилазе из чињенице да довођење дистрибутивне мреже на описану

локацију кошта приближно 2 милиона динара. Посматрано са друге стане, трошкови набавке и монтаже фотонапонског панела, акумулаторске батерије и регулатора напона не премашују износ од 60.000 динара. Развојем технологије у домену искоришћења обновљивих извора енергије, у будућим годинама може се очекивати још нижа цена поменутих напонских компоненти.

Коришћењем фотонапонског панела и акумулаторске батерија за потребе напајања моделом предложене метеоролошке станице обезбеђено је стално напајање једног оваког система за праћење метеоролошких параметара. Овако креирано напајање, поред сталног снабдевања метеоролошке станице електричном енегријом, обезбеђује мобилност исте. Метеоролошка станица се може пребацити са једне локације на другу без додатних трошкова адаптације система напајања. У случају коришћења дистрибутивне електро мреже, мобилност метеоролошке станице условљена је локацијом електро мреже или проширењем електро мреже до саме локација на којој је потребно поставити метеоролошку станицу, што доводи до додатних трошкова реализације.

6. Процес преноса параметара

Процес прикупљања метеоролошких параметара, као што је описано у претходном поглављу, заснива се на коришћењу метеоролошке станице креиране за потребе коришћења на производним површинама. Основна идеја приликом креирања модела метеоролошке станице било је њено пројектовање за рад у систему од више метеоролошких станица и једне базне станице. На овакав начин, метеоролошке станице биле би постављене на одређеним локацијама на којима би вршиле мерење метеоролошких и просторно-временских параметара. Основном полазном претпоставком је дефинисано да локација базне станице није условљена локацијама метеоролошких станица.

Удаљени приступ и слање података од метеоролошке станице до базне станице представља додатни проблем који захтева посебну анализу у циљу креирања адекватног решења. Као једно од потенцијалних решења, намеће се коришћење бежичних сензорних мрежа у виду преноса података путем радио фреквенције или преноса података путем *GSM* мреже. Примена једне или друге технологије, као и евентуална примена хибридног решења које би обухватило коришћење обеју технологија условљена је распоредом метеоролошких станица, њиховом међусобном удаљеношћу и конфигурацијом терена на којем су метеоролошке станице постављене.

6.1 Пренос параметара радио фрекфенцијом

Процес преноса параметара је могуће реализовати применом хардверских решења која ће омогућити коришћење радио фреквенције. Теоријски посматрано, у оквиру већ поменутих хардверских уређаја на метеоролошкој станици, потребно је додати радиофреквентни примопредајник. Базна станица, са друге стране, мора да поседује радиофреквентни примопредајник. Уколико се ради о систему у коме се налази већи број метеоролошких станица, у зависности од примењеног хардверског решења, метеоролошке станице и базна станица морају се поставити тако да задовољавају одређену топологију. Сама топологија је условљена конфигурацијом терена и максималном линијом видљивости између базне станице и метеоролошких станица.

Стандард који је развијен последњих неколико година и који је нашао интензивну примену у паметној пољопривреди и другим областима безичних сензорних мрежа, јесте *ZigBee* стандард. *ZigBee* је бежична телекомуникациона технологија развијена као отворени глобални стандард који треба да одговори на јединствене захтеве ниске цене реализације и мале потрошње енергије бежичних *PAN* (енг. *Personal Area Network*) мрежа. *ZigBee* је скуп комуникационих протокола вишег слоја, дефинисан од стране групе компанија под називом *ZigBee Alliance*, утемељених на IEEE 802.15.4 стандарду за *WPAN* (енг. *Wireless Personal Area Network*), базираном на радио преносу. *ZigBee* алијанса је савез са више од 300 компанија, које имају за циљ развој бежичних мрежа које могу имати велики број уређаја и поседују могућност малог преноса података. Циљани пренос података је око 250 kb/s. Притом, овакве бежичне мреже треба да буду једноставне, јефтине, пауздане и да се карактеришу малим утрошком енергије. Поред свега наведеног, једна од основних особина коју *ZigBee* мрежа треба да има јесте могућност самоприлагођавања и преусмеравања порука, као и могућност рада у нелиценцираним слободним фреквентним опсезима. Нелинценцирани радио фреквентни опсези нису исти у свим деловима света, зато IEEE 802.15.4 користи три могућа опсега. Тиме се добија покривеност најмање једним опсегом, на било којој територији у свету. Поменута три фреквентна опсега су позиционирана на фреквенцијама од 868 MHz, 915 MHz и 2400 MHz. Предности које карактеришу употребу фреквентних опсега од 868 MHz и 915 MHz су већи број корисника, мања интерференција и мања апсорпција и рефлексија [115]. Са друге стране, фреквентни опсег на 2.4 GHz је далеко прихваћенији. Разлози веће употребе фреквентног опсега на 2.4 GHz су сама расположивост опсега, већи број канала, мања потрошња и већа брзина преноса.

Посматрано из домена урађаја који се могу наћи у једној оваквој мрежи IEEE 802.15.4 стандардом зависно од врсте сервиса који су понуђени, дефинисане су две класе уређаја:

- Потпуно функционални уређај (енг. *Full Function Device - FFD*) који се може користити у свим топологијама и може обављати све функције, односно може радити као *PAN* координатор, координатор или крајњи

мрежни уређај. *FFD* уређај је најчешће повезан са сталним извором напајања.

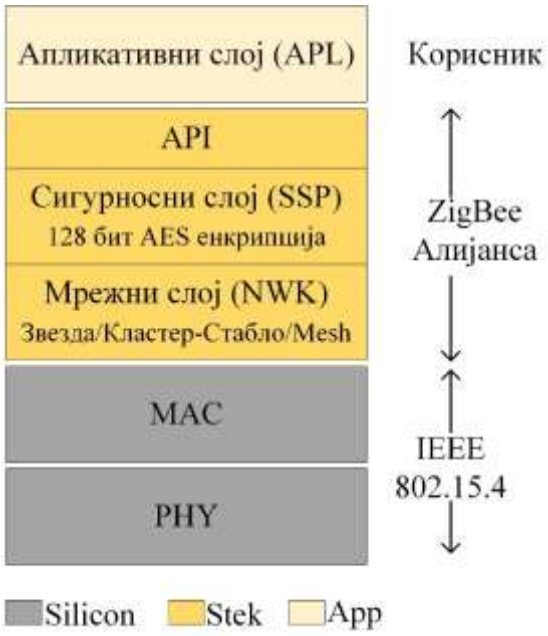
- Уређај са смањеном функционалношћу (енг. *Reduces Function Device - RFD* који се може користити само као мрежни уређај. Овакав уређај може се користити само у топологији звезде или као крајњи уређај у *peer-to-peer* топологијама. *RFD* уређај је енергетски самосталан, односно има неко властито аутономно напајање, те је као такав ограничен у енергетском смислу и функцијама у мрежи.

Пренесено на *ZigBee* мрежу, у мрежном слоју дефиисане су три гупе уређаја:

- *ZigBee* координатор је најспособнији уређај, представља корен стабла мреже и уређај који служи за спајање са другим мрежама. Свака мрежа има само један координатор. Координатор поседује могућност памћења информација о мрежи у којој се налази. Стављену паралелу са IEEE 802.15.4 мрежним уређајима координатор представља *FFD* уређај, па је самим тим још једна његова функционалност иницијализација мреже.
- *ZigBee* рутер. Овај уређај повећава физички домет мреже омогућавајући већем броју чворова спајање у мрежу. Као и координатор и рутер је увек *FFD* уређај.
- *ZigBee* крајњи чвор (енг. *End Device*) - садржи управо онолико функција колико му је довољно да комуницира са матичним чвором (било координатором или рутером). *ZigBee* крајњи чвор обезбеђује функције пасивног мода рада или такозваног мода мировања, чији је основни задатак уштеда батерије. На овакав начин крајњи чвор захтева коришћење најмање количине енергије, те је самим тим јефтинији од координатора и рутера. Крајњи уређај је према својој структури углавном *RFD* уређај, мада у појединим случајевима може бити и *FFD* уређај.

На основу чињенице да је *ZigBee* протокол базиран на IEEE 802.15.4 стандарду, може се усвојити да овај протокол, такође базиран на *OSI* (енг. *Open System Interconnection*) моделу, с тим да дефинише само оне слојеве који су важни за остваривање функционалности у жељеном подручју. Организација *ZigBee* протокол стека дата је на слици 27.

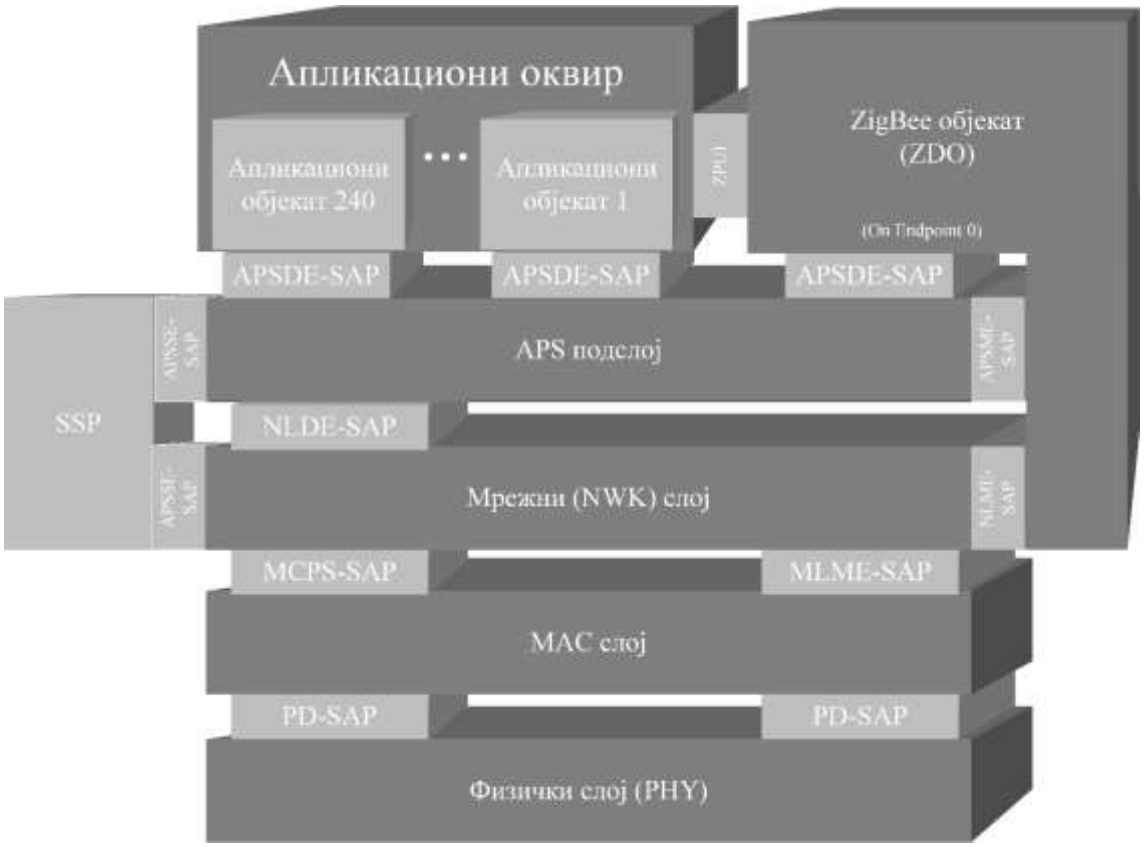
Практично *MAC* (енг. *Medium Access Control*) подслој као и PHY (енг. *Physical Layer*) слој представљају слојеве IEEE 802.15.4 стандарда. Слојеви вишег нивоа који се налазе изнад *MAC* слоја представљају слојеве *ZigBee* стандарда. С тим у вези, дефинисано је да се *ZigBee* стандард заснива на мрежном (енг. *Network Layer - NWK*) слоју, сигурносном (енг. *Security Service Provider - SSP*) подслоју и апликативном (енг. *Application Layer - APL*) слоју.



Слика 27: ZigBee протокол стек модел

Практично, сигурносни подслој се може посматрати и као део мрежног слоја, с обзиром да су на мрежном слоју имплементирани механизми за енкрипцију целих мрежних оквира. Након *ZigBee* слојева налази се ниво који представља корисничке апликације које користе *ZigBee* стандард за пренос података. *ZigBee* стек архитектура приказана је на слици 28. Сваки од слојева у прокол стеку нема никакве информације о слоју који се налази изнад њега. Сваки виши слој се може посматрати као мастер слој који управља слојем који се налази испод њега. На овакав начин, сваки од слојева изграђује виши ниво софистицираности базирајући га на темељима претходног слоја. На основу свега овога, може се рећи да се *ZigBee* стандард не уклапа у потпуности у седмослојни *OSI* модел, али исто тако садржи неке *исте* елементе као што су поменути *PHY*, *MAC* и *NWK* слојеви. Слојеви 4-7 OSI модела (транспорт, сесија, презентација и апликација) су *wrap-*

ovani у *APS* (енг. *Application Support*) подслој и *ZDO* (енг. *ZigBee Device Object*) слојеве у оквиру *ZigBee* модела. Између сваког од слојева налазе се SAP (енг. *Service Access Point*) тачке. *SAP* обезбеђује *API* (енг. *Application Programing Interface*) који врши раздвајање послова датог слоја од слојева који се налазе изнад и испод посматаног слоја. Као и IEEE 802.15.4 спецификација и *ZigBee* користи два *SAP* приступа по сваком слоју, при чему је један намењен подацима, а други за управљање. Примера ради сва комуникација са подацима до и од мрежног слоја пролази кроз *NLDE-SAP* (енг. *Network Layer Data Entity Service Access Point*).



Слика 28: Изглед ZigBee архитектуре

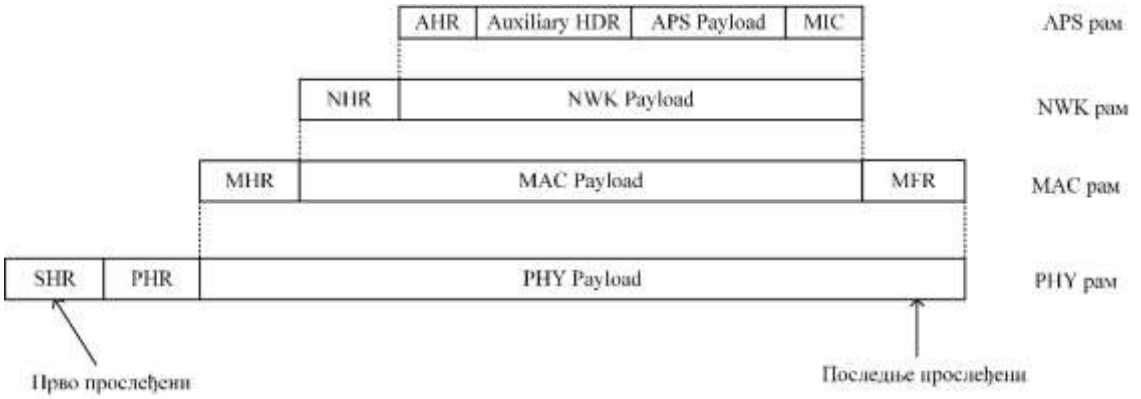
Најнижи протокол слој у *ZigBee* безичној мрежи приказаној на слици 27 и слици 28 је IEEE 802.15.4 физички слој у ознаци *PHY*. Задатак овог слоја је дефинисање структуре хардверских уређаја, као и фрекветног опсега бежичне комуникације. Практично, на физичком слоју се врши активирање и деактивирање примопредајника, мерење јачине енергије сигнала, мерење параметара квалитета преноса или откривање снаге сигнала, индикација квалитета

линка који мрежни слој користи за одабир слободних фреквентних канала, провера да ли је канал слободан, као и одабир канала који ће се користити за примање и слање података. Посматрано из угла фрекветног опсега и броја канала који се могу користити, у сваком од фреквентних опсега физички слој IEEE 802.15.4 протокола, као што је раније напоменуто, дефинише три слободна фреквентна опсега:

- Фреквентни опсег на 868 MHz је фреквентни опсег који се примењује на територији Европе. Овај фреквентни опсег садржи један канал и поседује могућност преноса података брзином до 20 kb/s.
- Фреквентни опсег на 915 MHz је фреквентни опсег који се примењује на територији Северне Америке. Овај фреквентни опсег садржи до 10 канала, при чему је брзина преноса 40 kb/s.
- Фреквентни опсег на 2,4 GHz је фреквентни опсег који се примењује у целом свету. Овај фреквентни опсег садржи до 16 канала, при чему је брзина преноса до 250 kb/s.

Теоријски посматрано примена *ZigBee* уређаја на фреквентном опсегу од 2,4 GHz може имати потенцијалне проблеме због чињенице да велики број безичних мрежа ради управо на овом опсегу. Приликом практичне реализације, како не би дошло до преклапања између *ZigBee* и *wifi* комуникације, потенцијалне сметње се избегавају бирањем различитих радних канала. Овако организован физички слој представља везу између уређаја за пренос података и MAC слоја.

Пренос податка и команди између различитих уређаја обавља се у форми пакета. Основна структура пакета је дата на слици 29. Пакети физичког слоја састоје се од три компоненте: заглавље синхронизације у ознаци *SHR* (енг. *Synchronization Header*), физичко заглавље у ознаци *PHR* (енг. *PHY header*) и физичко оптерећење (енг. *PHY payload*). *SHR* омогућава пријемнику синхронизацију и закључавање у оквиру тока битова. *PHR* садржи информације о дужини рама, док је *PHY payload* обезбеђено од виших слојева и укључује податке или команде за пријемни уређај.



Слика 29: Структура ZigBee пакета

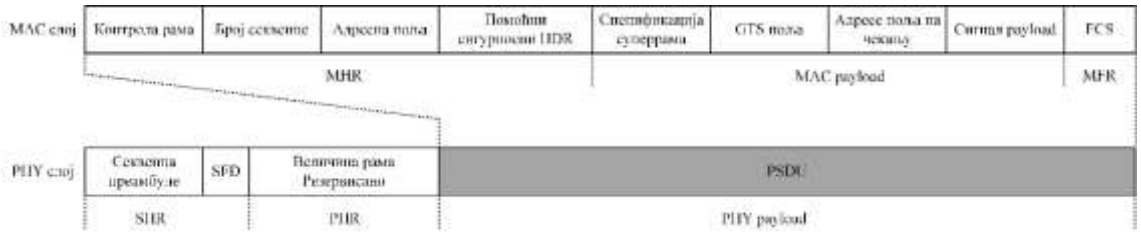
Задатак IEEE 802.15.4 *MAC* слоја је комуникација са физичким слојем. Тачније, овај слој обезбеђује интерфејст између физичког и мрежног слоја. Практично, овај слој је задужен за генерисање и синхронизовање мреже, покретање координатора и генерисање *PAN Id*-а (енг. *Personal Area Network Identinfier*). Такође, *MAC* слој је задужен за генерисање сигнала и синхронизацију уређаја у оквиру сигнала. Још један од задатака овог слоја јесте и коришћење *CSMA/CA* (енг. *Carrier Sense Multiple Access with Collision Avoidance*) алгоритма. Самим тим, ово је алгоритам који омогућава да у околностима присуства велике количине шума, односно када је однос сигнал/шум мали, *ZigBee* даје одличне перформансе. Примена овог алгоритма заснива се на ослушкивању стања на каналу. Ако је канал заузет, *ZigBee* уређај неће слати информације, све док канал не буде слободан.

Постоје случајеви када се не користи наведени *CSMA/CA* алгоритам. Овакви случајеви обухватају слање *beacon*-ова, који се шаљу по фиксном временском распореду. Исто тако и поруке о потврди пријема не користе *CSMA*. Уз овај механизам и *DSSS* (енг. *Direct Sequence Spred Spectrum*) технику раширеног спектра *ZigBee* уређаји комуницирају робусно и ефективно и у присуству великих интерференција, нарочито када раде у опсегу од 2,4 GHz који користи и *wifi*. Такође, у оквиру *MAC* слоја врши се и слање порука у вези потврђивања пријема (енг. *Acknowledger frame delivery - ACK*), као и састављање и растављање рамова порука [116]. *MAC* оквир који се преноси другим уређајима као *PHY payload* има три секције као што се може видети на слици 29. *MAC* заглавље у ознаци *MHR* садржи информације, какве су адресирање и безбедност. *MAC payload* има величину променљиве дужине (укључујући и нулту дужину) и

садржи команде или податке. *MAC* футер у ознаци *MFR* садржи 16-битну *FCS* (енг. *Frame Check Sequence*) секвенцу за верификацију података. Што се тиче формата оквира који се дефинише у оквиру *MAC* слоја IEEE 802.15.4 стандардом, дефинисана су четири могућа формата рамова:

- рам сигнала (енг. *Beacon frame*);
- рам за податке (енг. *Data frame*);
- рам потврде преноса (енг. *Acknowledge frame*);
- MAC командни рам (енг. *MAC command frame*).

Изглед структуре рама сигнала дат је на слици 30. Целокупан *MAC* рам се користи као паулоад у оквиру *PHY* пакета. Садржај *PHY payload*-а се односи на *PSDU* (енг. *PHY Service Data Unit*).

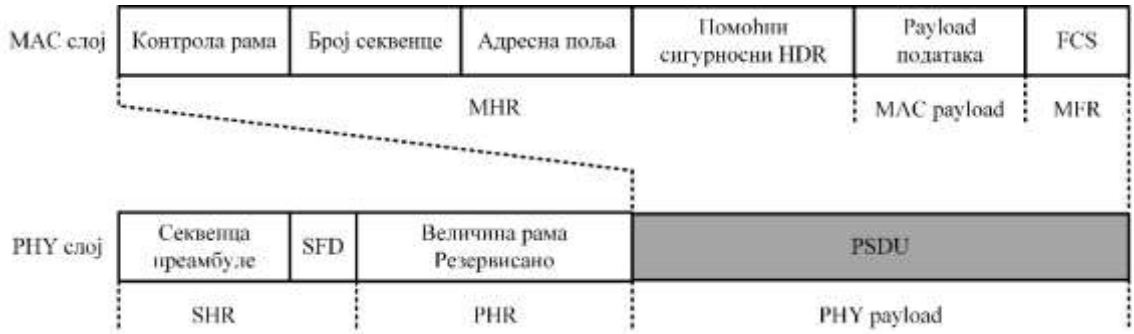


Слика 30: Формат MAC рама за сигнал

Рам сигнала се не користи само за потребе синхронизације уређаја у мрежи, већ се користи и као координатор како би се конкретном уређају у мрежи ставило до знања да постоје подаци на координатору који чекају на њега. Уређај ће по свом нахођењу контактирати координатора и затражити пренос података. Овакав поступак се назива индиректна трансмисија. Посебно поље у *MAC payload*-у садржи адресу уређаја који има податке који чекају на координатора. Сваки пут када уређај прими овакав сигнал, он ће проверити поље за адресу уређаја који је на чекању како би видео да ли постоје подаци који су на чекању.

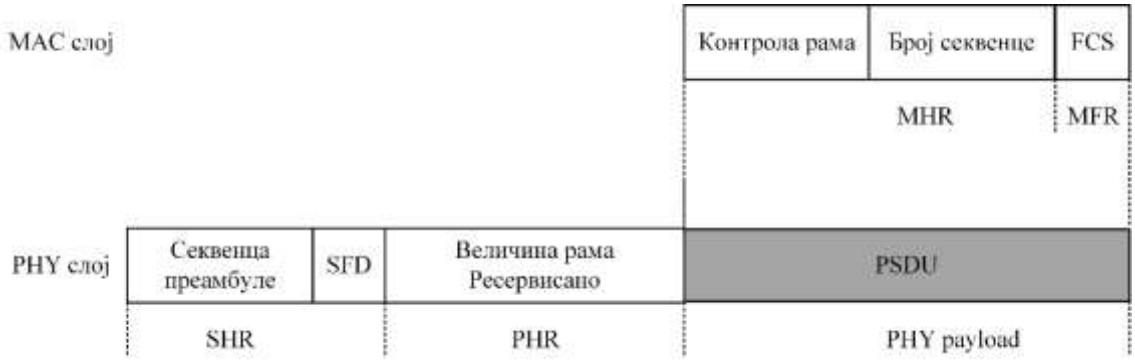
Beacon payload поље је опционо поље који се може користити од стране мрежног слоја и које се може преносити даље са *Beacon* рамом. Пријемник користи FCS поље како би извршио проверу на могуће грешке у примљеном оквиру.

Изглед структуре рама за података дат је на слици 31. *Payload* података је обезбеђен од стране мрежног слоја. Подаци у *MAC* корисничком носиоцу се називају MAC јединице за услуге (енг. *MAC Service Data Unit*). Поља у оквиру овог рама су слични раму сигнала, са изузетком суперрама, *GTS*-а и чињенице да поља за адресе чекања нису присутна у оквиру *MAC* рама за податке. MAC рам за податке означава се *MPDU* (енг. *MAC Protocol Data Unit*) и постаје *PHY* payload.



Слика 31: Формат MAC рама за податке

MAC рам за механизам потврде преноса је најпростији формат оквира. Структура овог рама је приказана на слици 32. Овај MAC рам не преноси било какав *MAC* payload. Рам потврде преноса се шаље од стране једног уређаја до другог како би се потврдио успешан пријем пакета података.



Слика 32: Формат MAC рама за потврду пријема пакета података

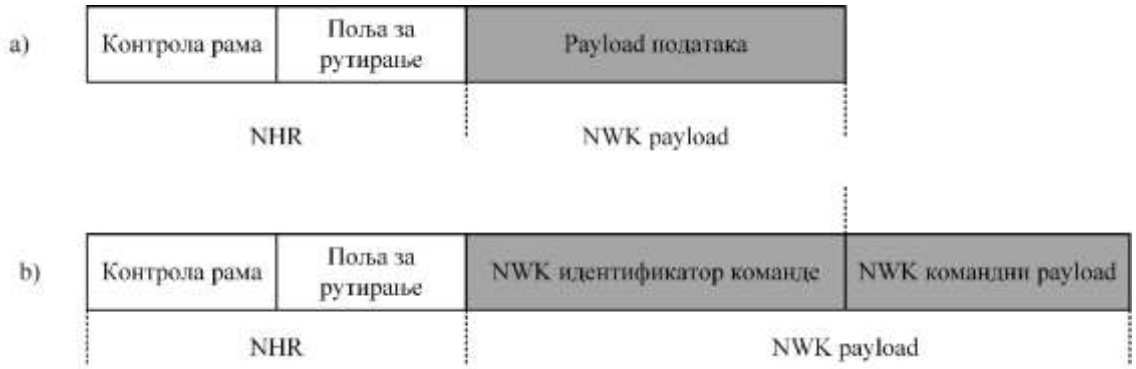
MAC командни рам приказан на слици 33 користи се за прослеђивање команди, као што су захтев за приступ мрежи или напуштање мреже. Тип командног поља одређује тип команде која ће бити послата, односно захтев за приступ мрежи или захтев за подацима. *Payload* команде садржи сопствену команду. Целокупни *MAC* командни рам је смештен у оквиру *PHY payload*-а као *PSDU*.

Слика 33: Формат MAC командног рама

Мрежни слој трећи је по реду слој у *ZigBee* и IEEE 802.15.4 протокол стеку. Тачније, овај слој се може посматрати као први слој вишег нивоа, који је уједно и први слој *ZigBee* протокол стека. Основни задатак мрежног слоја је правилно формирање топологије мреже, конфигурација свих уређаја у мрежи, додавање и избацивање чворова из мреже, достављање поруке на тачно одредиште, правилно адресирање, детекција суседних и правих путева између два чвора у мрежи, као и прослеђивање и примање података од апликативног и *MAC* слоја. Посматрано из угла могућих топологија код *ZigBee* уређаја најчешће се срећу топологија звезде, топологија стабла и мрежаста (енг. *Mesh*) топлогија [117].

Топологија звезде се састоји од једног координатора и једног или више крајњих уређаја. У овој топологији уређаји могу једино комуницирати путем координатора, односно не могу комуницирати директно. Основна карактеристика топологије стабла је да има један чвор на врху (корен) на који су, по принципу гранања, повезани остали чворови. Порука унутар ове мреже путује уз стабло и низ стабло. У *mesh* топологији, чворови су повезани са другим чворовима, тако да вишеструки путеви повезују сваки чвор. Веза између чворова се динамички ажурира и оптимизује преко софистициране уграђене *mesh* табеле за рутирање. *Mesh* мреже су децентрализоване. Сваки чвор је у стању да самостално истражује мрежу. Такође, када чвор напусти мрежу, mesh топологија омогућава чворовима да реконструишу путање рутирања на основу нове структуре мреже. Карактеристике *mesh* топологије и *ad-hoc* рутирање осигуравају већу стабилност приликом промене задатака или појаве одређених проблема на појединим чворовима [118].

Уколико се посматра присуство *ZigBee* уређаја у оквиру поменутих мрежних топологија, уочава се да је за све топологије заједничко да морају да имају најмање два главна уређаја, а то су координаторски чвор и крајњи чвор. Рутер као један од *ZigBee* уређаја је опциони, што значи да се он не мора наћи у мрежи, већ се користи само у посебним случајевима када је потребно извршити проширење мреже. Мрежни слој поседује и сет команди за сигурносне механизме. На овакав начин цео *payload* мрежног оквира је екриптован. *ZigBee* стандардном су дефинисна два могућа формата мрежног рама, и то: рам података и комадни рам. Приказ формата ова два мрежна рама је дат на слици 34.



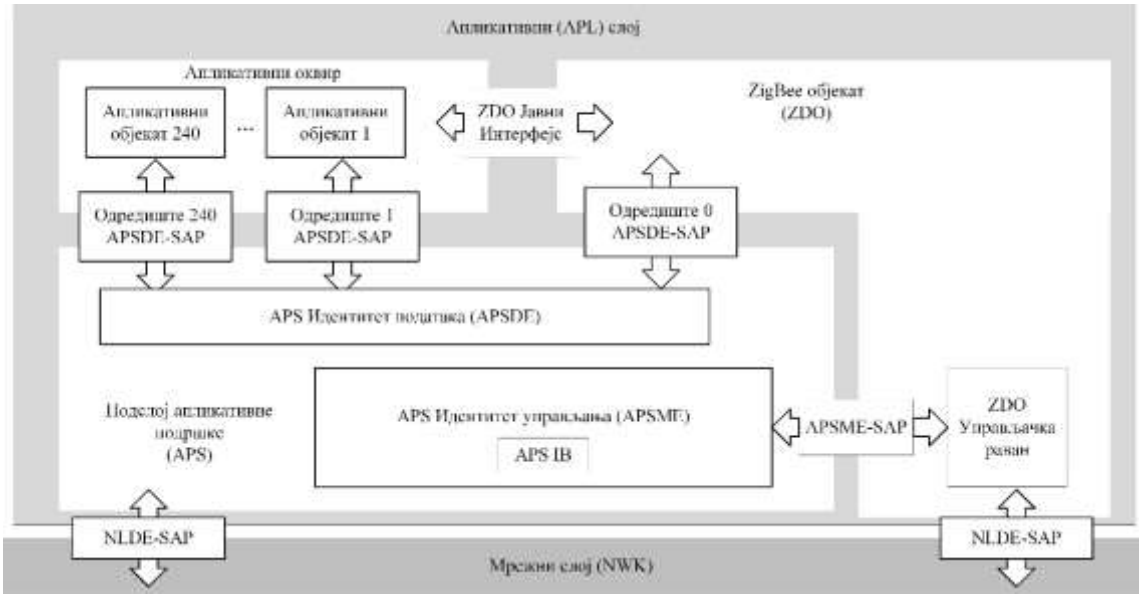
Слика 34: Формат а) рама података, б) командног рама

Слој мреже дефинише коришћење десет могућих команди. Свака од могућих команди носи различит идентификатор. Такође, свака од могућих десет команди има другачији формат поруке који носи конкретне информације потребне за извршење конкретне команде [116].

Сигурноси подслој (енг. *Security Service Provider - SSP*) пружа сигурносне механизме за мрежни и апликативни слој. Овај подслој врши енкрипцију података и може се посматрати као део мрежног слоја, што се може видети на слици 27. У највећем броју случајева *ZigBee* уређаји користе 128-битне кључеве како би имплементирали сигурносне механизме. IEEE 802.15.4 стандардом је подржано коришћење *AES* (енг. *Advanced Encryption Standard*) механизма за енкрипцију података и комуникације. Кључ може бити повезан или са мрежом, тако да се користи у оба *ZigBee* слоја и *MAC* слоју, или путем везе остварене пре-инсталацијом, споразумом или транспортом.

Успостављање кључева за повезивање је засновано на главном кључу који контролише коресподенцију везе кључева. У оваквом механизму се иницијални главни кључ мора добити путем неког сигурног медија као што је транспорт или пре-инсталирање, с обзиром да сигурност целе мреже зависи од њега. Веза и главни кључеви су видљиви само слоју аплиација. Различите услуге користе различите једносмерне варијације кључа како би се избегло цурење и остали сигурносни ризици.

Последњи и уједно и највиши ниво ZigBee протокола је апликативни слој. Овај слој се према слици 28 може посматрати као скуп више појединачних секција. Тачније, апликативни слој се састоји од три секције: апликативни оквир (енг. *Application Framework*), *ZigBee* објекат (енг. *ZigBee Device Object - ZDO*) и апликативни подслој подршке (енг. *Appication Support sublayer - APS*). Организација секција у оквиру апликативног слоја се може видети на слици 35.



Слика 35: Организација апликативног слоја у ZigBee мрежи

Апликативни оквир, као прва од секција, има задатак да обезбеди спецификацију преко које је могуће креирати профил на *ZigBee* протокол стеку. Апликативни оквир у *ZigBee* мрежи је окружење у којем су апликативни објекти хостовани како би контролисали и управљали слојевима протокола на *ZigBee* уређају. Објектне апликације развијају произвођачи у случајевима када су уређаји развијени за различите апликације.

Као што се може видети са слике 35, на једном уређају се може наћи до 240 апликација. Апликативни објекти користе *APSDE-SAP* како би вршили размену података (слање и примање) између равноправних чланова. Сваки од објеката има јединствену одредишну адресу (одредишна адреса 1 до 240). Одредишна адреса 0 је заузета и користи се од стране *ZDO*-а. За потребе слања поруке свим апликативним објектима користи се одредишна адреса 255. Одредишне адресе омогућавају да више уређаја дели исти радио.

ZigBee објекат као друга секција обезбеђује интерфејс између *APS* подслоја и аплокативног оквира. *ZDO* садржи функционалности које су заједничке у свим апликацијама чији се рад базира на *ZigBee* протокол скету. На пример, у оквиру овог дела врши се одређивање улоге коју конкретни уређај има у оквиру *ZigBee* мреже (координатор, рутер или уређај). Поред овога, у овом делу се врши и стартовање и/или одговарање на захтеве за повезивањем или на захтеве за откривањем руте. Такође, у овом делу се успостављају и сигурне везе између мрежних уређаја. *ZDO* користи примитиве за обављање својих дужности и приступа *APS* подслоју користећи *APSME-SAP*. *ZDO* јавни интерфејс се користи за интеракцију са апликативним оквиром. Ова раван чини размену информација између мрежног и апликациног слоја могућом. ZigBee стандард нуди могућност коришћења профила апликација приликом развоја апликације. Профил апликације је скуп споразума о формацијама за одређене апликације и поступке обраде. Употреба профила апликације омогућава дању интероперабилност између производа развијених од стране различитих произвођача за одређену апликацију. Уколико два произвођача користе исти профил апликације како би развили своје производе, производ једног произвођача ће моћи да комуницира са производима произведеним од стране другог произвођача као да су оба произведена од стране истог произвођача.

Апликативни подслој подршке представља трећу секцију апликативног слоја. Ова секција обезбеђује интерфејс између мрежног и апликативног слоја. *APS* под слој слично свим нижим слојевима подржава два типа сервиса: подаци и управљање. Сервис *APS* података је обезбеђен *APS* ентитетом података (енг. *APS Data Entity - APSDE*) и приступа му се кроз *APSDE - SAP*.

Способности управљања се нуде кроз APS ентитет управљања (*APSME*) коме се приступа кроз *APSME-SAP*. Овај подслој показује одређене сличности са *TCP/IP* протоколом које се огледају у оквиру пружања подршке апликативним услугама. Са друге стране, за разлику од *TCP/IP* протокола апликативни подслој за подршку не поседује могућност обезбеђивања контроле тока података.

Примењено на моделом описану метеоролошку станицу, одабир хардверског решења за радио примопредајник мора бити такав да користи исто напајање као и *Raspberry Pi* рачунар. Такође, одабрано хардверско решења мора да поседује техничке могућности повезивања са *Raspberry Pi* рачунарем у циљу адекватног слања и пријема података и управљачких сигнала. Теоријски посматрано, практична реализација се огледа у набавци два истоветна радио фрекветна модула, од којих ће један бити постављен на метеоролошкој станици, док ће други бити постављен у оквиру базне станиц. Такође, у зависности од конфигурације терена и максималне пропагације сигнала у оквиру линије видљивости постоји и могућност увођења додатних појачавача снаге сигнала. Појачавачи снаге сигнала се примењују уколико је потребно пренети сигнал на већу раздаљину од максималне раздаљине коју основни предајник може пренети. Такође, појачавачи се користе у случајевима када је терен брдовит или када постоје препреке које утичу на пропагацију самог сигнала. Појачавач појачава ослабљену снагу сигнала и потребно га је поставити на месту до кога сигнал емитован са предајника долази ослабљен. Различите фамилије радио фреквентних модула погодних за реализацију слања података са метеоролошке станице доступне су од стране компаније *Digi International*[3]. Свака од фамилија карактерише се различитим функционалностима уређаја које обухвата. Фамилије *Xbee* радио фрекветних уређаја ове компаније приказане су табели 11.

Табела 11: XBee фамилије RF модула

Назив фамилије	Фреквенца	Начин склапања	Протокол
XBee ZigBee	2.4GHz	TH/SMT	ZigBee
XBee 802.15.4	2.4GHz	Through-Hole	802.15.4
XBee S2C 802.15.4	2.4GHz, 900MHz	TH, SMT	802.15.4
XBee DigiMesh 2.4	2.4GHZ	Through-Hole	DigiMesh

[3] Digi International, https://www.digi.com/

Свака од наведених фамилија *XBee* радио фреквентних модула садржи већи број уређаја од који се сваки може применити на решавање конкретних проблема креирања радио фреквентне бежичне мреже. Основне разлике између наведених фамилија уређаја односе се на *form faktor*, као и протокол на коме се базира њихов рад. Што се тиче фреквенце на којој ови уређаји раде, може се приметити да све наведене фамилије уређаја раде на фреквенци од 2.4 GHz, што је најучесталија фреквенца бежичних мрежа у нелиценцираном фреквентном домену. Такође, може се уочити да *XBee S2C* 802.15.4 фамилија радио фреквентних модула може радити и на 900 MHz. Што се тиче технологије склапања неке од фамилија радио фреквентних уређаја, засноване су на старијем принципу у ознаци *TH* (енг. *Through-Hole*), док друге користе новији принцип склапања у ознаци *SMT* (енг. *Surface-Technology).* Притом *XBee S2C* 802.15.4 користи обе технологије у зависности од тога за коју фреквенцу се креира дати уређај.

Одабир одговарајућег уређаја за потребе реализације слања података од моделом описане метеоролошке станице до базне станице захтева детаљнију анализу перформанси уређаја који припадају поменутим фамилијама. Најбитније техничке карактеристике се огледају у брзини преноса података, потрошњи енергије, максималном домету, могућности повезивања и комуникације са *Raspberry Pi* уређајем итд. Поређење задовољења потребних техничких карактеристика дато је у табели 12.

Брзина преноса података свих наведених фамилија *RF* модула износи 250 kb/s. Уколико се ради о серијском преносу података код свих *RF* модула изузев фамилије XBee 802.15.4., брзина преноса може се повећати до 1 Mb/s. Што се тиче домета сигнала, упоређене техничке карактеристике односе се на рад у спољашњем окружењу, при чему постоји линија видљивости између два модула. Ово значи да између два *RF* модула нема физичких препрека које би могле узроковати сметње, као ни додатних извора који би узроковали интерференцију. Стварни домет RF модула варира у зависности од преносне снаге, оријентације предајника и пријемника, висине пријемне антене, временских услова, извора сметњи у околини примопредајника и терена између пријемника и предајника.

Табела 12: Анализа перформанси XBee фамилија RF модула

Фамилија	Категорија	Домет [m]	Осетљивост пријемника [dBm]	Број дигиталних U/I pinova	Број канала	Напајање [V]	Max потрошња [mAh]
XBee ZigBee	Digi XBee S2C ZigBee	1200	-100/-102 boost mode	15	16	2,1-3,6	59
	Digi XBee-PRO S2C ZigBee	3200	-101		15	2,7-3,6	120
	Digi XBee S2D ZigBee Thread Ready	1200	-100/-102 boost mode		16	2,1-3,6	45
XBee 802.15.4	Legacy Digi XBee S1 802.15.4	100	-92	8	16	2,8-3,4	50
	Legacy Digi XBee-PRO S1 802.15.4	1600	-100			2,8-3,4	215
XBee S2C 802.15.4	Digi XBee S2C 802.15.4	1200	-100/-102 boost mode	15	16	2,1-3,6	45
	Digi XBee-PRO S2C 802.15.4	3200	-101			2,7-3,6	120
XBee DigiMesh 2.4	Digi XBee S2C DigiMesh 2.4	1200	-100/-102 boost mode	15	16	2,1-3,6	45
	Digi XBee-PRO S2C DigiMesh 2.4	3200	-101		15	2,7-3,6	120

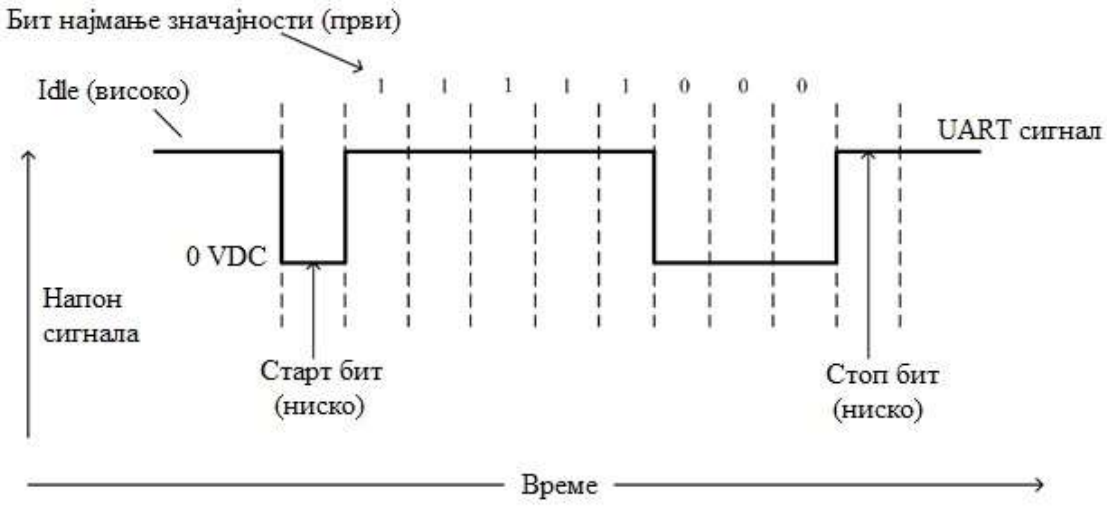
Такође, на максимални домет сигнала утичу и унутрашње и спољашње структуре као што су зидови, дрвеће, зграде, брда и планине. Категорије *RF* модула које у свом називу садрже реч *PRO* (*Digi XBee-PRO S2C ZigBee, Legacy Digi XBee-PRO S1 802.15.4, Digi XBee-PRO S2C 802.15.4*, и *Digi XBee-PRO S2C DigiMesh 2.4*) представљају уређаје који, као што се може видети из табеле 12,

имају боље техничке карактеристике у погледу максималног домета сигнала, а самим тим и максималне удаљености између два *RF* модула која се налазе у линији видљивости. Међутим, и поред добрих перформанси за рад у спољашњем окружењу, ограничавајући фактор је немогућност коришћења ових *RF* модула на територији Европе. Правилима којима се регулише коришћење радио фреквентног опсега на територији Европе није дозвољено коришћење оваквих уређаја.

Након елиминације поменутих категорија *RF* модула, а на основу техничких карактеристика преосталих фамилија и категорија *RF* модула за потребе реализације преноса података од моделом описане метеоролошке станице до базне станице одабрана је *Digi XBee S2C* 802.15.4 категорија која припада *XBee* S2C 802.15.4 фамилији *RF* модула. Уређаји који припадају овој категорији *RF* модула подржавају следеће топологије: један на један, један на више, *peer-to-peer* и *DigiMesh*. *XBee RF* модул приступа *host* уређају коришћењем серијског порта. Уређај може комуницирати са било којим логичким и напонски компатибилним *UART* (енг. *Universal Asynchronous Receiver/Transmitter - UART*) преко нивоа преводиоца на било који серијски уређај (примера ради преко *RS-232* или *USB* интерфејс плоче) или преко *SPI* (енг. *Serial Peripheral Interface*). Уређаји који поседују *UART* интерфејс повезују се директно на пинове *XBee ZigBee RF* модула. Слање података од уређаја до *UART-a XBee ZigBee RF* модула обавља се кроз *TH* пин, 4/*SMT* пин 4 *DIN* као асихрони серијски сигнал. Када уређај не врши пренос података, сигнал треба да буде постављен у стању *idle high*.

Како би се остварила серијска комуникација *UART* мора на оба уређаја (микроконтролер и *XBee ZigBee RG* modul) бити подешен компатибилно. Компатибилна подешавања обухватају брзину преноса, парност, старт битове, стоп битове и битове података. Подешавање поменутих вредности се врши унапред дефинисаним командама. Сваки бајт података састоји се од старт бита (ниски бит), 8 битова података (бит најмање значајности је први) и стоп бита (високо). На слици 36 је приказан шаблон серијског преноса битова кроз уређај. Практично, слика приказује *UART* пакет података 0x1F (децимални број 31) током преноса кроз уређај.

Посматрано са друге стране, *XBee ZigBee RF* модул подржава *SPI* комуникацију у *slave* моду. Тако, *slave* мод прима сигнал такта и податке од мастер уређаја и враћа податке мастер уређају. Комунукација између *slave* и мастер уређаја се обавља кроз унапред дефинисане четири команде. У основној конфигурацији оба *UART* и *SPI* порта су конфигурисана за серијски порт операцију. У овом случају серијски подаци иду кроз *UART*, све док *host* уређај не потврди омогућавање серијске комуникације са *slave*-ом слањем $SPI\_\overline{SSEL}$ сигнала. Након овога, сва серијска комуникација се извршава само на *SPI* интерфејсу све док се не обави ресет.



Слика 36: Пренос UART пакета података кроз уређај

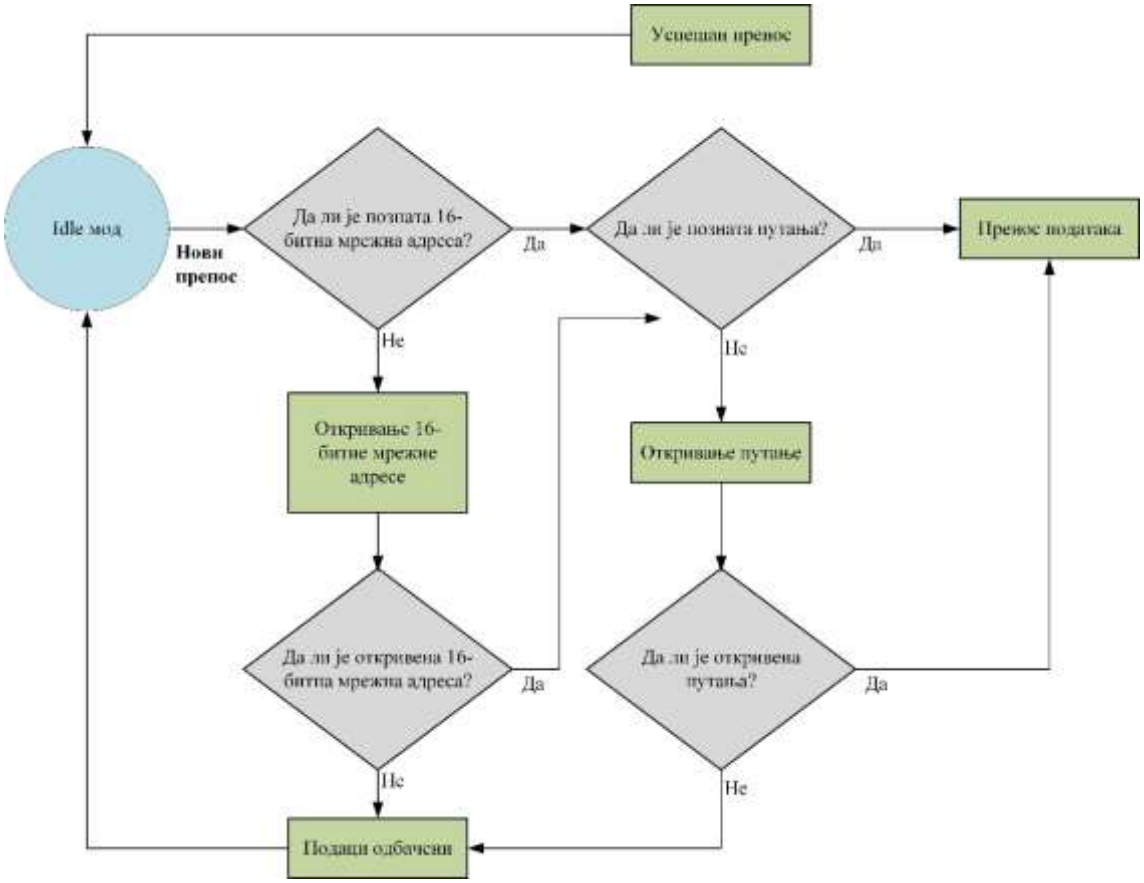
Период рада *XBee ZigBee RF* модула се може посмарати кроз неколико модова. Основни модови рада у којима се *XBee ZigBee RF* модул може наћи су:

- Мод преноса (енг. *Transmit mode*), у случајвима када су серијски подаци у оквиру серијског бафера спремни за паковање;
- Мод пријема (енг. *Recive mode*), у случајевима када су валидни RF подаци примљени кроз антену;
- Командни мод (енг. *Command Mode)*, у случајевима када је командна секвенца послата. Овај мод није доступан са паметним енергетским софтверима или приликом коришћења SPI порта;
- Пасивни мод или мод мировања (енг. *Sleep mode);*

- Idle моде, у случајевима када уређај не обавља нити пренос нити пријем података.

Пре самог процеса преноса података *RF* модул осигурава да су 16-битна мрежна адреса, као и путања до одредишног чвора у мрежи одређене. Уколико 16-битна мрежна адреса и путања до одредишног чвора у мрежи нису познате покреће се протокол њиховог одређивања. Уколико пак уређај са одговарајућом мрежном адресом није пронађен, долази до одбацивања пакета података. Уређај врши пренос података након одређивања путање. Уколико се поступак одеђивања путање заврши неуспехом, уређај одбацује пакете. На слици 37 је приказана секвенца мода преноса. Након обављеног преноса *ZigBee* података од једног до другог чвора, одредишни чвор шаље потврду пријема на мрежном нивоу назад до чвора пошиљаоца користећи већ успостављену путању. Пакет потврде пријема указује пошиљаоцу да је одредишни чвор примио пакет података. Уколико пошиљалац не добије потврду пријема паката података, покреће се ретрансмисија података. Постоје посебни случајеви у којима је могуће да одредишни чвор успешно изврши пријем пакета, а да притом пошиљалац не добије мрежну потврду пријема. У оваквом случају пошиљалац ће извршити ретрансмисију података, што може узроковати мултиплицирање пакета података на одредишту. Недостатак *XBee* модула је управо нефилтрирање дуплицираних пакета. Решење оваквог проблема се огледа на апликативном нивоу, где ће апликација садржати функционалности за обраду пристиглих пакета.

Мод пријема представља основни мод XBee ZigBee RF модула. Практично, уколико уређај не врши пренос података он се налази у моду пријема. Уколико одредишни чвор прима валидне RF пакете, онда овај чвор врши пренос података у свој серијски преносни бафер. Командни мод је стање у коме *firmware* интерпретира карактере које прима као команде. На овакав начин је кориснику дата могућност модификације конфигурације уређаја помоћу параметара које може да подеси користећи *AT* команде. Примера ради, уколико је потребно прочитати тренутну вредност параметра или подесити било који параметар *XBee ZigBee RF* модула користећи овај мод, корисник мора послати одговарајућу *AT* команду [119].



Слика 37: Секвенца мода преноса

Свака од *AT* команди почиње са *AT* након чега следе два карактера која идектификују команду, а за њима опционе конфигурационе вредности. Командни мод је доступан на *UART* интерфејсу за све оперативне модове. Током употребе *SPI* интерфејса не може се приступити командном моду.

Пасивни мод или мод мировања омогућава *XBee ZigBee RF* модулатору превођење у стање у коме ће знатно умањити утрошак енергије. Овај уређај подржава мод мировања пинова и циклични мод мировања. Током трајања овог мода уређај је скоро потпуно искључен и није у могућности да шаље или прима податке све до момента док не изађе из овог стања. Код мода мировања пинова уређај улази у мод мировања након пин транзиције. Што се тиче цикличног мода мировањ,а дефинише се фиксно време трајања овог мода након кога се уређај враћа у мод пријема.

Одабрани *XBee RF ZigBee* модул има 15 *GPIO* пинова. Намена пинова дефинисана је листом пинова при чему листа зависи од конфигурације уређаја.

Одређени пинови се могу користити за потребе серијске комуникације. Повезивање ових *RF* модула са микроконтролерима, као и њихово напајање може се обавити управо коришћењем *GPIO* пинова. Како се у случају моделом описане метеоролошке станице *Raspberry Pi* рачунар користи као микроконтролер потребно је обавити повезивање са овим уређајем. Повезивање одабраног *RF* модула са *Raspberry Pi* рачунаром може се обавити на два начина. Први начин огледа се у повезивању *GPIO* пинова *RF* модула и *GPIO* пинова *Raspberry Pi рачунара*. Овакво повезивање захтева коришћење додатне адаптер плоче. Оваквим начином повезивања поред коришћења додатног хардвера јавља се заузеће *GPIO* пинова на *Raspberry Pi* рачунару. Практично, оваквим начином је елиминисана могућност повезивања метеоролошких и просторно-временских сензора са *Raspberry Pi* уређајем путем истих *GPIO* пинова. Други начин повезивања подразумева коришћење *Uni4 XBee/ZigBee* адаптер плоче са *USB* интерфејсом. На овакав начин се врши ослобађање *GPIO* пинова *Raspberry Pi* рачунара како би се исти користили за друге намене, док се *XBee* модул путем *GPIO* пинова повезује са *USB* адаптером. У циљу реализације моделом описане метеоролошке станице за случај преноса података путем радио сигнала практичније је користити други начин повезивања.

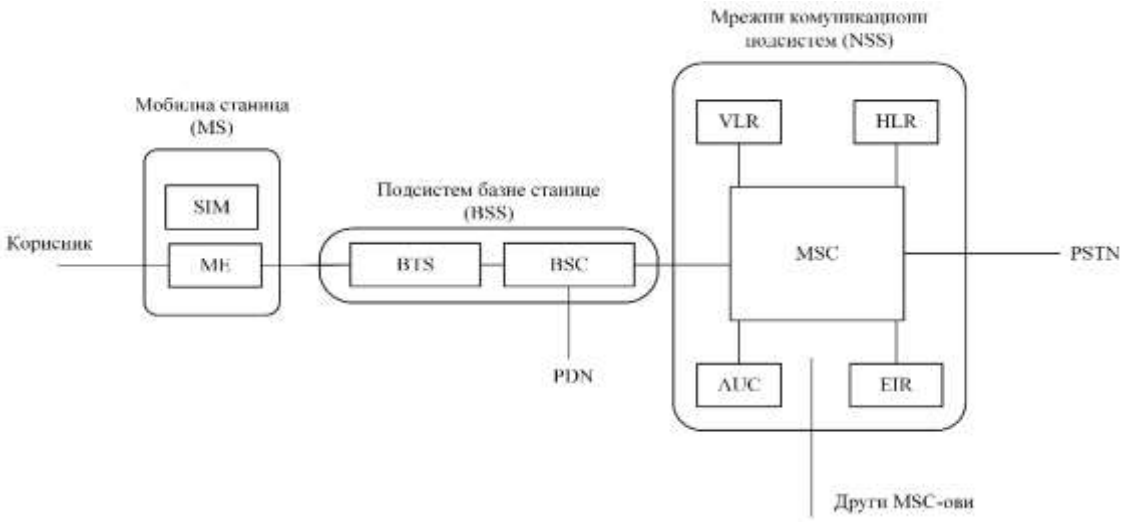
Поменути *USB* адаптер напаја се са 500 mA при напону од 3,3 V. Потрошња овог уређаја се може редуковати стављањем истог у мод мировања током периода у коме се дати уређај не користи. Посматрано из угла реализације мода мировања код *XBee ZigBee* модула применио би се цикличан мод мировања. Сходно томе, уређај би се из мода мировања у мод пријема постављао у фиксно дефинисаним временским размацима током дана. Након пријема пакета података од *Raspberry Pi* рачунара и прослеђивања истих путем радио фреквентног опсега *XBee ZigBee* модул би се враћао у мод мировања, чиме би се вршила уштеда енергије. Притом, значајно би се смањило оптерећење *Raspberry Pi* уређаја с обзиром на чињеницу да не би сви сензори и уређаји радили истовремено, што би смањило могућност оштећења *Raspberry Pi* рачунара услед преоптерећења или прегреања. Такође, овакав начин рада елиминише потребу за уградњом расхладног система рачунара.

6.2 Пренос параметара GSM/GPRS мрежом

Пренос параметара са једног мобилног уређаја до другог мобилног уређаја може се обавити коришћењем *GSM* (енг. *Global System for Mobile Communication*) или *GPRS* (енг. *General Packet Radio Service*) мреже. Генерално посматрано, мобилна мрежа је слична фиксној телефонији. Основна разлика се огледа у могућностима неограничене мобилности претплатника која је остварена коришћењем бежичног интерфејса са системом. *GSM* је практично европски стандард за дигиталне мобилне системе заснован на коришћењу три различита фреквентна опсега: 900 MHz, 1800 MHz и 1900 MHz. Основне предности ове технологије се огледају у могућностима остваривања међународног саобраћаја, високом квалитету преноса говорног сигнала, већој сигурности преноса информација и могућностима које овај стандард отвара када је у питању имплементација различитих других сервиса. *GSM* је организован у три главна дела:

- мобилна станица (енг. *MS – Mobile Station*);
- подсистем базне станице (енг. *BSS- Base Station Subsystem*);
- мрежни и комутаторски подсистем (енг. *Network and Switching Subsystem*).

Поједини елементи сваког од наведених далова *GSM*-а, као и њихова међусобна веза се могу видети на слици 38 која представља референтну архитектуру *GSM*-а.



Слика 38: Референтна архитектура GSM-а

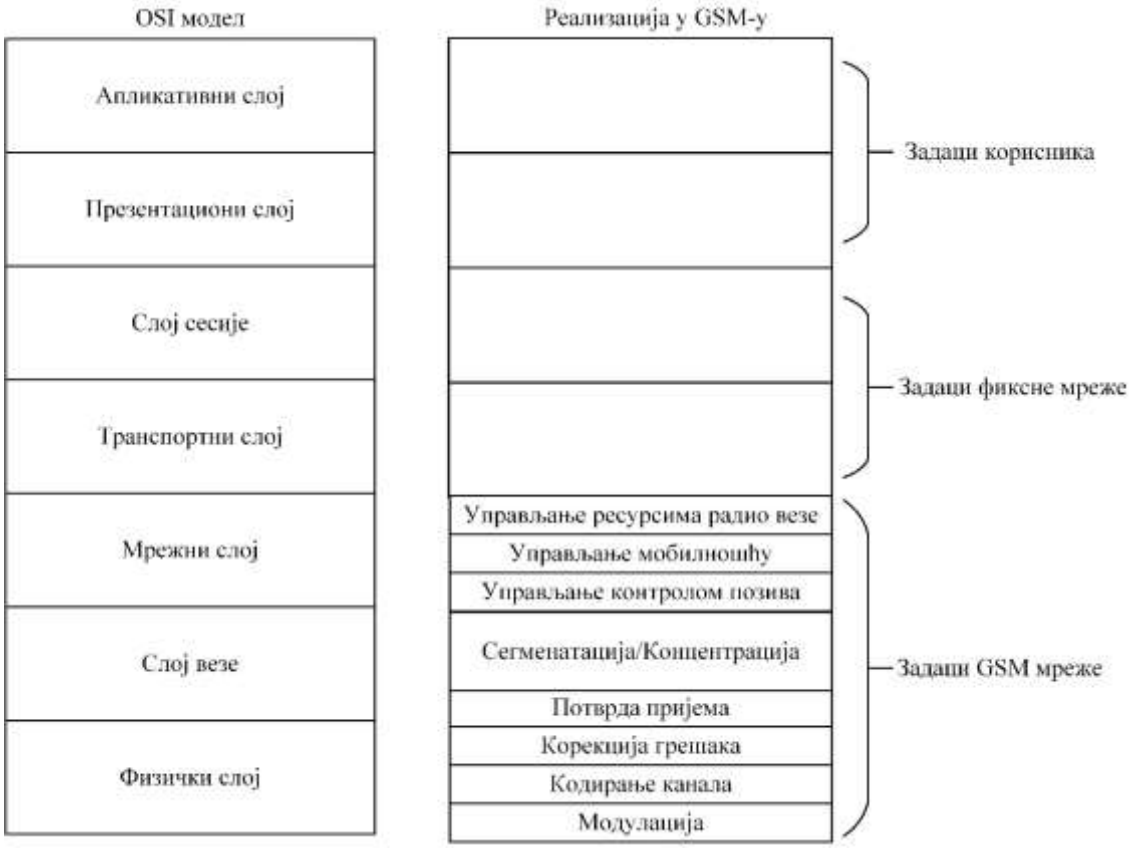
Мобилна станица, као што се може видети на слици 38, чине *ME* (енг. *Mobile Equipment*) и *SIM* (енг. *Subscriber Identity Module*) [120]. ME, у скаду са називом, представља опрему набављену од стране корисника. Овако набављена хардверска опрема омогућава имплементацију протокола којим се остварује интерфејс мобилне станице или мобилне једнице са корисником, као и бежични интерфејс са *BSS*-ом. *SIM*, као други елемент мобилне станице, представља добро познату картицу коју, такође, набавља корисник и која се користи у процесу идентификације спецификација које се односе на корисникову адресу и тип сервиса који се опслужује. *SIM* картица садржи личне податке о сваком кориснику који му обезбеђују да оствари приступ коришћењу већег броја корисних апликација.

Подсистем базне станице, као што је приказано на слици 38, се састоји од два елемента: базног припопредајног подсистема (енг. *Base Transceiver Subsystem - BTS*) и контролера базне станице (енг. *Base Station Controller - BSC*). Задатак *BTS*-а јесте остваривање физичке комуникације са *MS*-ом коришћењем етра као комуникационог медијума. Етар као комуникациони медијум је релативно непоуздан, па се морају обезбедити механизми успоставе сигурне комуникације. *BTS* је опремљен предајником, пријемником и сигналном опремом. Ова опрема је лоцирана у центру ћелије, управо на оном месту где се налази антена *BSS*-а. *BSC* као други градивни блок *BSS*-а представља мали комутатор који је задужен за фреквентно адресирање, као и *handover*-ом између *BTS*-а у оквиру *BSS*-а. На овакав начин, задатак *BSS* јесте комуникација путем етра, омогућавање мобилности корисницима, као и повезивање са ожичаном инфраструктуром коришћењем поузданијих жичаних протокола.

Мрежни и комутаторски подсистем представља основу рада мреже остваривањем комуникације са другим мрежама независно од тога да ли су друге мреже бежичне или жичане. *NSS* је, самим тим, најсофистицираније блок *GSM* мреже и састоји се од једног хардверског *MSC* (енг. *Mobile Switching*) блока и четири софтверска елемента: *VLR* (енг. *Visitor Location Register*), *HLR* (енг. *Home Location Register*), *EIR* (енг. *Equipment Identification Register*) и *AUC* (енг. *Authentication Center*).

MSC представља хардверску компоненту која је задужена за комуникацију са јавном телефонском мрежом коришћењем *SS*-7 сигналног протокола. Задатак *MSC*-а је обезбеђивање специфичне информације о статусу мобилних терминала. Ова информација се обезбеђује мрежи. *HLR* је у основи софтвер налик бази података чији је задатак ажурирање рачуна мобилног претплатника. Практично, у овој својеврсној бази података чувају се информације о адреси претплатника, типу услуге, текућој локацији и стању на рачуну. *VLR* је, такође, као *HLR* својеврсна база података која чува запис о локацији претплатника у оквиру области покривања *MSC*-а, односно о његовом кретању. *AUC* представља различите алгоритме који се служе за идектификацију и криптовање претплатника. Како различите класе *SIM* картица имају уграђене различите алгоритме за аутентификацију задатак овог центра се састоји у томе да прикупља све ове алгоритме и обезбеди *NSS*-у неомеран рад са различитим терминалима који потичу из различитих географских подручја. Последњи софтверски елемент *EIR* представља и својеврсну базу података која се користи за управљање идентификацијом мобилне опреме са аспекта кварова и крађа мобилне јединице.

Овако комплексан систем, као што је *GSM*, захтева прецизно планирање и организацију, како у домену дефиниције тако и у домену имплементације. Додатну сложеност овог система узроковала је потреба преноса података путем *GSM* мреже која је омогућена *GPRS*-ом. Уколико је спецификација *GSM* мреже упореди са организацијом слојева у *OSI* референтном моделу, може се уочити да GSM прати прва три слоја OSI модела. Практично, први и други слој *GSM* модела одговарају физичком и *data link* слоју *OSI* модела. Трећи слој *GSM* модела одговара мрежном слоју *OSI* модела, што се може видети на слици 39. Физичке карактеристике преносног медијума су специфициране у најнижем слоју. Посматрано у контексту *GSM* радио везе, ова дефиниција не укључује само радио фреквентни носиоц и *GMSK*, већ и тачан временски распоред преноса који је потребан због употребе *TDMA* (енг. *Time-Division Multiple Access*). Овај слој обезбеђује и методе за коректно преношење битова, додавањем редундантних битова за корекцију грешака помоћу конволуционог кодирања и шири пренос података помоћу преклапања.



Слика 39: OSI референтни модел примењен на GSM

Други слој *GSM* модела одговара дата линк слоју и састоји се од интелигентног ентитета који је задужен за сигуран пренос порука са подацима између мобилне и базне станице. Овај слој обезбеђује везу између мрежног слоја који се налази изнад и физичког слоја који се налази испод овог слоја. Такође, овај слој обезбеђује детекцију грешака и корекцију пакета података примљених са физичког слоја. На овакав начин се од примаоца захтева потврда пријема пакета података како би се обезбедило поновно слање оних пакета који нису примљени. На пријемној страни другог слоја врши се реконструкција порука из примљених рамова, на основу којих се формира потврда пријема. Како би се обезбедила адекватна веза, што је основни задатак овог слоја, користи се *LAPDm* (енг. *Link Access Protocol for the ISDN D-channel*) протокол за канал *D*. Овај протокол је најпре коришћен код *ISDN*-а (енг. *Integrated Services Digital Network),* док је његова модификована верзија примењена на *GSM*, отуда и „*m*“ у називу протокола.

Разлог ове модификације је ниска брзина преноса података у ваздушном простору. Супротно од *LAPD* протокола коришћеног код *ISDN*-а, *LAPDm* протокол не садржи *start* и *end* флегове за потребе синхронизације, као ни идентификатор крајње тачке када је опрема крајњег корисника у питању. Изостанак оваквог идентификатора оправдан је с обзиром да се ради о *point-to-point* комуникацији. Такође, *LAPDm* протокол не захтева коришћење секвенце за проверу фрејма (енг. *Frame Check Sequence - FCS*), која се иначе користи за идентификацију грешака у преносу. Практично, као што је раније наведено, у оквиру *GSM* мреже контрола грешака се извршава од стране физичког слоја.

Трећи слој *GSM* модела представља мрежни слој *OSI* референтног модела и одговоран је за управљање успостављеном везом и активностима повезаним са радио мрежом. У оквиру трећег слоја *GSM*-а могу се издвојити три подслоја: управљање ресурсима радио везе (енг. *Radio Ressource Management - RR*), управљање мобилношћу (енг. *Mobility Management - MM*) и управљање контролом позива (енг. *Call Control - CC*).

Подслој управљања ресурсима радио везе је задужен за успостављање и одржавање стабилне и непрекидне комуникационе путање између *MSC* и *MS* путем које се преносе сигнали и кориснички подаци. Преноси су део одговорности *RR* слојева. Већина функција је контролисана од стране *BSC, BTS* и *MS*, иако се неки од њих врше од стране *MSC*-а (нарочито пренос између више *MSC*-а). Подслој за управљање мобилношћу поред аутентификације и процедура кодирања задужен је и за очување података о локацији. Са друге стране, подслој контроле позива је задужен за подешавање позива на захтев корисника. Самим тим, овај слој обавља послове организације услуга усмерених ка позивима, модификацијама и проверама конфигурације додатних сервиса, као и послове обезбеђивања размене кратких порука.

Поред преноса позива и гласа у оквиру *GSM* мреже реализован је и пренос података путем *GPRS*-а. Основна разлика између *GSM*-а и *GPRS*-а у погледу преноса је у томе што *GSM* користи технику комутације кола (енг. *Circuit Switching*), док *GPRS* користи пакетну комутацију (енг. *Packet Switching*).

Посматрано у домену комуникације између уређаја, прималац пакета може бити други корисник, као и било који сервер уређај на Интернету. Како би се остварио пакетни пренос *GPRS* података *GSM* мрежом извршене су одређене модификације. Како би се код пакетне комутације остварило рутирање порука потребно је уградити скуп комутаторских чворова. На овакав начин се креира привидно паралелна мрежа унутар које се налазе оперативни чворови. Овако креирана мрежа састоји се од две главне функционалне целине: *gateway* према спољашњој мрежи за пренос података и чвора чија је улога опслуживање корисника. *GGSN* (енг. *Gateway GPRS Support Node*) представља тачку приступа спољној мрежи за пренос података и у стању је да рутира пакете ка текућој локацији мобилног корисника. Информација о локацији мобилног корисника коме се опслужује пакет добија се од стране *HLR*-а коме *GGSN* мора имати приступ. Други чвор који представља надоградњу *GSM*-а и опслужује потребе мобилне станице означава се *SGSN* (енг. *Serving GPRS Support Node*). *SGSN* је задужен за управљање радом мобилне станице са тачке гледишта мобилности. Шифровање пакетно оријентисаних порука обавља се помоћу *SGSN*-а. На овакав начин се обе функционалности могу комбиновати у један јединствени чвор.

Главна предност пакетно-комутираног преноса се огледа у флексибилности које овај пренос нуди, како за потребе претплатника тако и за операторе. Када корисник жели да преузме податке, он користи укупно расположиву пропусност везе. За случај да постоје и други корисници, они ће, у зависности од индивидуалних захтева или нивоа приоритета, делити расположиву пропусност. Код преноса са комутацијом кола, у току позива или преноса података, ресурс (линија) се додељује једном кориснику независно од тога да ли се у датом тренутку и по тој вези преноси нека информација или не. То обично значи да постоји процедура успостављања везе (енг. *call set-up procedure*), тј. процедура доделе ресурса. Код оваквог преноса ресурси остају резервисани за потребе корисника у току целог времена преноса. Задатак мрежних елемената се састоји у томе да одржавају везу, обављају *handover*, ако је неопходно, и меморишу информацију о трајању разговора ради наплате. Овакав начин преноса је у потпуности различит у односу на пакетно-комутирану везу код које се ресурси деле и не постоји гаранција да се у току преноса неће јавити кашњење поруке.

До кашњења долази када два корисника покушавају да приступе истом ресурсу, при чему један од њих има приоритет, тако да ће порука од другог бити закашњена. Друга разлика се састоји у адресирању, тј. дефинисању изворишта и одредишта поруке. У недостатку *call set-up* процедуре мрежа мора тачно да зна како да одреди локацију примаоца текућег пакета података. Због овога се сваком пакету придружује додатна информација о адресирању. С обзиром на то да се сваки временски слот може наизменично користити између већег броја корисника, наплаћивање рачуна не може бити базирано на времену, већ се користи нови модел заснован на количини података која се преноси за потребе сваког од претплатника. Основне предности које *GPRS* у процесу преноса података нуди углавном су последица коришћења технике *packet switching*, а то су:

- Директна размена података уз коришћење Интернет или Интранет компанија. Директна размена је могућа, с обзиром да су уређаји у стању да манипулишу пакетним преносом податка;
- Пакети креирани од стране једног корисника могу се бежично преносити у периоду од неколико временских слотова (енг. *bundling*);
- Пренос података не мора да буде у континуалним временским слотовима, шта више, расположиве слотове могуће је истовремено користити од стране већег броја корисника (прво шаље један, затим други, након тога трећи корисник итд.);
- Чак и када корисници не врше примопредају пакета података, они и даље остају повезани/доступни са *LAN*-ом њихове компаније, а да при томе не користе било који ресурс;
- Корисници представљају само алоциране ресурсе и њима се приступа само када је то потребно;
- *GPRS* се имплементира са *GSM* стандардом, тако да нема потреба за коришћењем нових фреквенција.

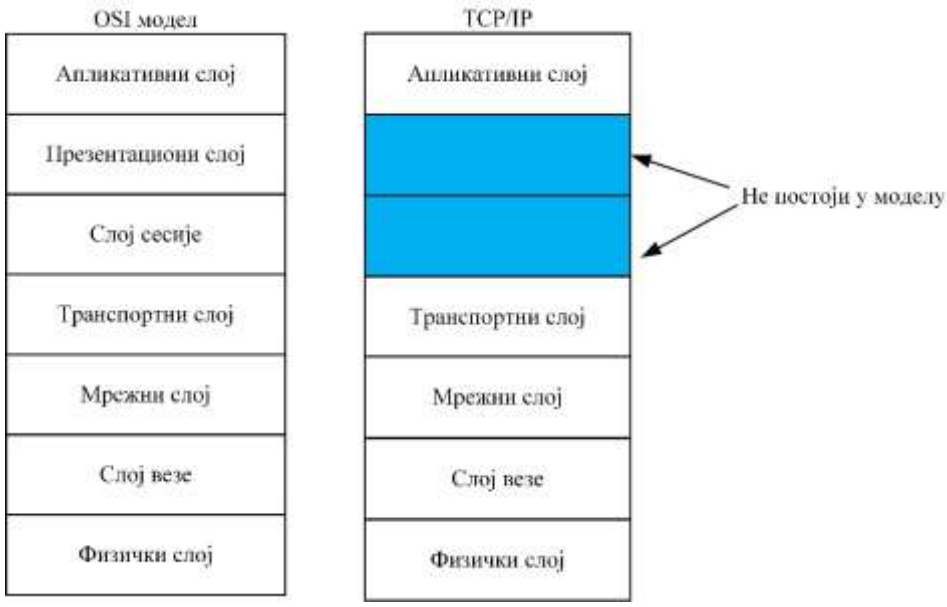
У комуникационим мрежама које покривају већа географска растојања, као што су *GSM* мрежа, *GPRS* мрежа и Интернет, комуникација између пошиљаоца и примаоца се остварује преносом података кроз мрежу коју чине чворови

посредници. Када се ради од *GSM* мрежи, чворови посредници могу бити базне станице, док се рецимо код интернет везе за чворове посреднике сматрају рутери. Практично, задатак чворова посредника није интерпретација садржаја поруке као ни значења података, већ сами пренос података од једног до другог чвора који се налазе на путањи од предајника до пријемника. Овакав концепт преноса података се назива комутација пакета. Концепт дефинише да се поруке преносе у оквиру кратких блокова о којима је и раније било речи, а који се најчешће називају пакети података. Како би се остварило успешно слање пакета података од предајника до пријемика, односно од пошиљаоца до примаоца, дефинисани су посебни протоколи који регулишу сами пренос порука и података.

Главни протокол на мрежном нивоу *OSI* модела реализованог на *GSM* мрежу је *IP* (енг. *Internet Protocol*). Поред чињенице да је ово главни протокол, интернет протокол свакако није једини протокол дефинисан у оквиру мрежног слоја. Поред овог протокола постоји још неколико помоћних протокола као што су *ARP*, *RARP*, ICMP, *IGMP* итд. Интернет протокол је одговоран за комуникацију између рачунара (тзв. *host-host* комуникација). Као протокол мрежног слоја, интернет протокол испоручује поруку од рачунара који представља пошиљаоца до рачунара који представља примаоца поруке. Међутим, ово је непотпуна испорука, јер често није довољно само испоручити поруку одредишном рачунару, већ је треба и предати одговарајућем процесу на одредишном рачунару који ће је прихватити и обрадити. Другим речима, коначно одредиште поруке није рачунар, већ апликациони процес на одредишном рачунару (тзв. процес-процес комуникација). Управо овај последњи корак у испоруци порука представља одговорност протокола транспортног слоја, као што су *UDP* і *TCP*.

Основни скуп протокола који се користи за контролу преноса дефинисан је у оквиру TPC/IP (енг. *Transmission Control Protocol/Internet Protocol*) модела. *TCP/IP* је референтни модел који се користи на Интернету. TCP/ *IP* референтни модел је развијен пре *OSI* модела, тако да се слојеви ова два модела не поклапају у потпуности [121]. *TCP/IP* модел чини пет слоја: физички, слој везе, интернет слој, транспортни и апликациони. Тачније, може се рећи да је *TCP/IP* модел сачињен од четири слоја (слој везе, интернет слој, транспортни и апликациони), при чему

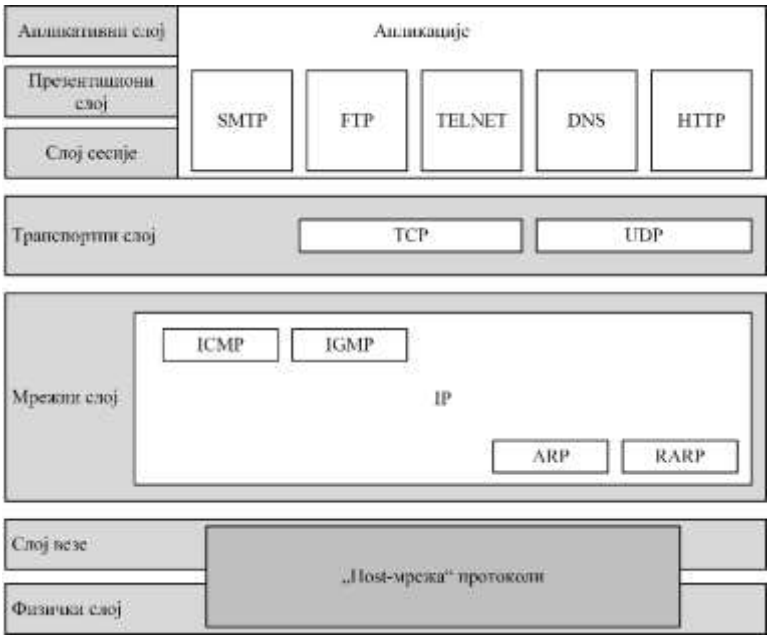
се додавањем физичког слоја овај модел може посматрати као хибридни модел. *TCP/IP* се само спорадично бави најнижим слојевима (физичким и слојем везе). Заједно, ова два слоја се третирају као „хост-мрежа“ слој. *TCP/IP* не намеће неке посебне захтеве који се тичу ових слојева. Претпоставља се да мрежа поседује протоколе који покривају функције тих слојева, а нагласак ставља на слој мреже, транспортни и апликациони слој. Мрежни и транспортни слој одговарају трећем и четвртом слоју *OSI* модела. Међутим, код *TCP/IP* модела на транспортни слој се директно наставља апликациони слој, који обухвата функционалност три вршна слоја *OSI* модела. Изглед слојева *TCP/IP* модела у односу на слојеве *OSI* модела може се видети на слици 40.



Слика 40: Однос између OSI и TCP/IP модела

TCP/IP је хијерархијски скуп протокола сачињен од интерактивних, али не и обавезно међусобно независних модула од којих сваки остварује неку специфичну функцију. За разлику од *OSI* модела који дефинише које функције припадају ком слоју, слојеви *TCP/IP* модела садрже релативно независне протоколе који се могу комбиновати зависно од потреба система. Појам хијерархијски значи да је сваки протокол вишег слоја подржан од стране једног или више протокола нижег слоја. На слици 41 је приказана структура *TCP/IP* модела са протоколима разврстаним у слојеве који су преклопљени са одговарајућим слојевима *OSI* модела.

Као што је раније наведено на транспортном слоју, *TCP/IP* дефинише два протокола: *TCP* и *UDP* (енг. *User Datagram Protocol*). *UDP* протокол се може посматрати као веома једноставан бесконекциони протокол који је, притом, непоуздан. Овај протокол примењује се у случајевима када апликације, чији се пренос података базира на овом протоколу, не захтевају строгу контролу грешака које могу настати током преноса, као ни редоследа пристизања пакета података на одредиште. На пример, у групу апликација које за потребе комуникације и преноса података користе *UDP* протокол могу се уврстити апликације за пренос аудио и видео материјала. Код оваквих апликација акценат није на прецизој испоруци, већ на брзини испоруке пакета података. Управо из ових разлога, *UDP* протокол је једноставнији од рецимо *TCP* и *IP* протокола. *UDP* протокол удаљеним апликацијама омогућава размену енкапсулираних *IP* датаграма. Размена енкапсулираних *IP* датаграма се обавља без потребе успостављања конекције [122].



Слика 41: Структура протокола по слојевима TCP/IP модела

Овако конципиран *UDP* протокол апликацијама које учествују у комуникацији пружа основну функционалност размене података која се остварује на транспортном нивоу. Уколико се упореде *UDP* датаграми и *IP* датаграми, основна разлика која се може уочити јесу бројеви портова који се користе код *UDP* протокола.

Бројеви портова су уведени како би се предајна апликација приликом комуникације обратила тачно одређеној апликацији која се налази на одредишној машини. Основни недостатак *UDP* протокола се огледа у томе што *UDP* протокол не поседује механизме за контролу тока и контролу грешака у преносу, као ни могућност ретрансмисије након пријема лошег датаграма. Ово је уједно и сличност *UDP* протокола са *IP* протоколом, с обзиром да ни *IP* протокол не поседује поменуте механизме. Област примене *UDP* протокола се огледа код клијент-сервер конфигурације. У оваквој конфигурацији клијент шаље упит серверу. На задати упит клијент очекује конкретан и кратак одговор. Међутим, уколико упит не стигне до сервера или уколико се одговор од сервера изгуби током преноса, клијент након неког времена једноставно покуша поново. На пример, апликација којој је потребна информација о тачном времену може послати *UDP* датаграм са захтевом неком од сервера који пружају овакве информације. Након пристизања захтева, сервер ће одговорити *UDP* датаграмом у коме ће бити уписан тренутни датум и тренутно време. Како би се обавила оваква једноставна размена није потребно вршити било какве претходне припреме или успостављање конекције, довољно је разменити поменуте две поруке. Друга област примене *UDP* протокола су мултимедијалне апликације које раде у реалном времену: Интернет радио, Интернет телефонија, музика-на-захтев, видео конференције, видео-на-захтев и друге. Основна заједничка карактеристика оваквих апликација је *streaming* звука и/или видеа.

TCP је транспортни протокол конекционог типа који омогућава успостављање поузданог тока бајтова између две удаљене апликације. *TCP* обавља сегментацију тока бајтова на поруке које прослеђује интернет слоју. На страни одредишта, *TCP* реконструише ток бајтова и прослеђује га апликацији. За разлику од *UDP*-а, *TCP* је конекциони протокол. Овај протокол креира виртуелну конекцију између два удаљена процеса како би омогућио пренос података. Такође, *TCP* протокол врши контролу протока и контролу грешака [123]. Управо из ових разлога *TCP* протокол је прилагодљив променљивим карактеристикама Интернета, па самим тим нуди могућност одржавања стабилне и поуздане везе чак и у случајевима када дође до било каквог отказа у инфраструктури мреже. *TCP* протокол је оријентисан на ток и то га разликује од *UDP* протокола.

Код *UDP*-а, процес (апликациони програм) шаље поруке које имају тачно дефинисане границе. Свакој таквој поруци *UDP* додаје своје заглавље и испоручује је *IP*-у. Поруке се зову кориснички датаграми, а свака од њих у коначном облику постаје један *IP* датаграм. *UDP*, као ни *IP*, не виде било какву везу између датаграма. Са друге стране, *TCP* омогућава предајном процесу да шаље, а пријемном процесу да прима податке у виду континуалног тока бајтова. *TCP* креира окружење у којем изгледа као да су два процеса спојена неком имагинарном „цеви“, кроз коју теку њихови подаци. Практично, на овакав начин предајни процес генерише (уписује) ток бајтова, а пријемни процес конзумира (чита) бајтове из имагинарне „цеви“. *TCP* нуди услугу пуне дуплексне (енг. *full-duplex*) комуникације, где подаци могу да теку у оба смера у исто време. То значи да сваки *TCP* поседује оба бафера (предајни и пријемни), а сегменти се преносе у оба смера. *TCP* је поуздани транспортни протокол. Поузданост се постиже механизмом потврђивања. Пријемна страна потврђује примљене податке, а предајна поново шаље (ретрансмитује) податке који нису потврђени у неком дефинисаном времену. *TCP* нумерише све бајтове података који се преносе путем успостављене конекције. Нумерисање је независно у оба смера. Увек када добије бајт података од предајног процеса, *TCP* смешта бајт у предајни бафер и додељује му редни број. Нумерисање бајтова је кључна особина *TCP*-а, а која доминира свим аспектима протокола. Редни бројеви се користе за контролу протока и контролу грешака. *TCP* се, за разлику од *UDP*, бави контролом протока. Код *TCP* протокола пријемник је у могућности да контролише динамику којим предајник шаље податке. Контрола протока је неопходна како пријемник не би био претрпан подацима које предајник шаље исувише великом брзином. Захваљујући редним бројевима, *TCP* је у могућности да спроводи контролу протока до нивоа бајтова. Механизам за контролу грешака је неопходан да би се обезбедио поуздани пренос. Практично, задатак *TCP* протокола је да на пријемној страни реконструише првобитни ток бајтова, без обзира на евентуалне грешке у пакетима података, као и на евентулано дуплицирање или губитак пакета података. Иако се детекција грешака обавља на нивоу пакета података, концепт контроле грешака, као и контрола протока, је оријентисан на нивоу бајтова.

Како TCP протокол креира виртуелну конекцију и виртуелну путању између изворишне и одредишне апликације, оваква путања олакшава процес потврђивања пријема и ретрансмисију оштећених или изгубљених пакета података. Виртуелна конекција омогућава *TCP* протоколу коришћење услуга *IP* протокола, иако је *IP* протокол бесконекциони протокол, а *TCP* конекциони. Управо је суштина у томе да *TCP* конекција није физичка, већ виртуелна. Прецизније *TCP* протокол ради на вишем нивоу апстракције од *IP* протокола. Услуге *IP* протокола се користе само на нивоу испоруке појединачних пакета података пријемнику, док *TCP* протокол на виртуелном нивоу обавља контролу конекције. Уколико у процесу преноса дође до оштећења или губљења пакета података, *TPC* протокол иницира ретрансмисију. Са друге стране, уколико неки пакет података стигне ван редоследа, *TCP* протокол ће га привремено задржати све до момента док не стигну сви пакети податка који недостају. Процедуре за обезбеђивање успешног преноса података креиране од стране *TCP* протокола извршавају се тако да *IP* протокол није свестан истих. На основу свега наведеног, може се рећи да су потврда преноса података и ретрансмисија најзначајнији механизми *TCP* протокола, што га ставља испред других протокола за пренос података.

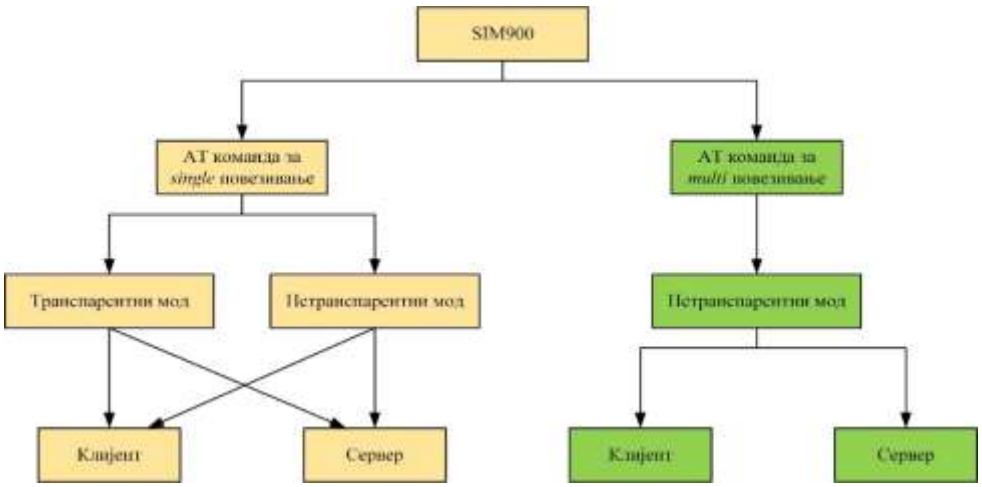
Пренесено на процес слања података од моделом описане метеоролошке станице до базне станице употреба *GSM/GPRS* мреже огледа се у коришћењу хардверског решења које се са једне стане може повезати са *Raspberry Pi* рачунарем, док се са друге стране може повезати са рачунарем који представља базну станицу. Практично, на страни метеоролошке станице један од параметара који утиче на одабир *GSM/GPRS* модема огледа се у реализацији потребног напајања које треба да буде у опсегу подржаног напајања од стране *Raspberry Pi* рачунара, с озбзиром на чињеницу да већина оваквих модема нема могућности спољашњег напајања, већ напајање добија са рачунара на који су повезани. Такође, један од параметара који утичу на одабир одговарајућег *GSM/GPRS* модема је и реализација комуникације између одабраног модема и Raspberry Pi рачунара, као и перформансе успоставе мрежне конекције и слања пакета података коришћењем *GSM/GPRG* мреже. *GSM/GPRS* модем на пријемној страни (базна станица), у зависности од техничких карактеристика рачунара који представља базну станицу, као и начина повезивања овог рачунара са *GSM/GPRS* модемом,

може бити исти *GSM/GPRS* модем као на изворишној страни (метеоролошка станица). Притом, реализација комуникације коришћењем два истоветна модема није обавезна, с обзиром на чињеницу да *GSM/GPRS* мрежи могу приступати различити уређаји. Уколико се ради о систему већег броја метеоролошких станица умрежених са једном базном станицом, одабрани *GSM/GPRS* модем базне станице мора поседовати могућност повезивања са већим бројем уређаја како би несметано могао да прима прослеђене пакете података. Како у највећем броју случајева базна станица представља персонални рачунар и како је њена локација у оквиру објекта, могућности напајања *GSM/GPRS* модема нису ограничене, као што је то случај са напајањем на страни метеоролошке станице. Такође, повезивање базне станице и *GSM/GPRS* модема може се обавити на више начина.

Данас је на тржишту могуће пронаћи различите *GSM/GPRS* модеме који нуде различите могућности повезивања и различите начине комуникације и слања података. Узимајући у обзир лимитиране могућности напајања, начин повезивања и рад у отвореним условима метеоролошке станице, једно од решења које се извојило јесте *SIM900-TTL UART GSM/GPRS* модел компаније *Rhydo Technologies (P) Ltd*[4]. Поменути *GPS/GPRS TTL UART* модем је заснован на *Quad-band GSM/GPRS* enginu у ознаци *SIM900A*. Овај engine ради на следећим фреквенцијама: 850 MHz, 900 MHz, 1800 MHz и 1900 MHz. Веома је компактан у погледу величине и коришћења у склопу *GSM* модема. Овај модем је дизајниран за рад у оквиру 3,3V/5V *TTL* кола, што му омогућава директно повезивање, како са микроконтролерима који раде на 3,3 V тако и са микроконтролерима који раде на 5 V. Повезивање на извор напајања се може остварити у опсегу од 5 V до 12 V. Пропусни опсег се може подешавати у опсегу од 9600 kb/s до 115200 kb/s. Сами пропусни опсег подешава се помоћу *AT* команди. Иницијално, модем је подешен на пропусни опсег од 9600 kb/s. Повезивање модема са микроконтролером засновано је на два проводника (*Tx, Rx*) за комуникацију и један проводник за напајање [124]. Овај *GSM/GPRS TTL* модем има интерни *TCP/IP* стек који омогућава успостављање интернет конекције путем *GPRS*-а. Погодан је, како за слање *SMS* порука тако и за коришћење од стране апликација за слање пакета података путем *TCP/IP*, као што је случај са метеоролошким станицом.

[4] Rhydo Technologies (P) Ltd, http://www.rhydolabz.com/

Уколико се посматра структура, *TCP/IP* протокол стека *SIM900* модема приказаног на слици 42, може се уочити да постоје два мода примене: међусобно повезивање два уређаја (енг. *Single connection*) и повезивање већег броја уређаја (енг. *Multi conncection*). Такође, процес преноса података *SIM900* модема може се одвијати у такозваном транспарентном моду и нетранспарентном моду. Уколико се ради о једној конекцији, *SIM900* може радити, како у транспарентном моду тако и у нетранспарентном моду, при чему *SIM900* може бити конфигурисан као *TCP/UDP* клијент или *TCP* сервер. Уколико се креира вишеструко повезивање, *SIM900* може радити само у нетранспрентном моду. У овом моду *SIM900* може радити као апсолутни *TCP/UDP* клијент, при чему као клијент може остварити максимално 8 конекција. Такође, у овом моду овај модем може бити конфигурисан као један *TCP* сервер, што омогућава да се седам *TCP/UDP* клијента конектује са њим. Постоји могућност и да се овако конфигурисан *TCP* сервер понаша као клијент, при чему може остварити седам конекција са једним удаљеним (енг. *Remote*) сервером. На овакав начин обезбеђено је проширење мреже, уколико је потребно међусобно повезати већи број уређаја.



Слика 42: Структура TCP/IP протокол стека SIM900 модема

Одабир *single* или *multi* конекције се обавља помоћу *AT* команди, при чему је подразумевани мод *single* конекција. Такође, подразумевани мод преноса података је нетранспарентни мод. Одабир између транспарентног и нетранспарентног мода се извршава слањем одговарајуће *AT* команде. Транспарентни мод преноса података *SIM900* модема обезбеђује остваривање

посебног мода података који служи за пријем и слање података путем *TCP/IP* апликаионе примене. Једном када је конекција путем транспарентног мода остварена, овај модем ће бити у моду података. На овакав начин се сви примењени подаци са серијског порта третирају као пакети података који касније требају бити прослеђени. На исти начин, сви примљени подаци се са удаљеног уређаја шаљу директно на серијски порт. Коришћење *AT* команди у транспарентном моду није могуће. Међутим, имплементиране су методе које омогућавају прелазак са мода података на командни мод и супротно [125]. Након преласка у командни мод, коришћење *AT* команди је опет могуће.

Како је раније наведено, *SIM900* може бити конфигурисан у три радна стања: *TCP* клијент, *UDP* клијент и *TCP* сервер. Поменута три стања *SIM900* модема су подржана, како у транспарентном моду тако и у нетранспарентном моду преноса података. Такође, ова три стања остварива су и у *single* и у *multi* моду конектовања уређаја. *SIM900* модул подржава *DNS* ауто парсирање, што могућава успостављање *TCP*/*UDP* конекције до удаљеног сервера директним коришћењем имена домена у оквиру *AT* команде. Потребни параметри *AT* команде су*: <мод>, <име домена>* и *<порт>*. Уколико се ради о *multi* конекцијама поред поменутих параметара потребно је навести и број конекције *<n>*. Слање података, било у *single* моду или *multi* моду повезивања, може се обавити на три начина: као слање података променљиве величине, као слање података фиксне величине и као временско слање. Исто тако, *SIM900* поседује методе обавештавања о томе колико је података послато од стране модула, а колико је примљено од стране удаљеног сервера током активне *TCP* конекције. Слање података променљиве величине захтева дефинисање максимлане величине пакате података која износи 1460 бајтова. Заправо, максимална величина података је променљива и одређена је од стране тренутне мреже која се користи. Како би се обавило овакво слање података, мора се утврдити колика је максимална величина пакета података дозвољена од стране мреже. У сваком случају, ова величина не може бити већа од поменутих 1460 бајтова. Слање података фиксне величине се обавља тако што се подаци шаљу аутоматски када величина улазних података достигне вредност унапред дефинисане променљиве *length*. Временско слање података је још један од аутоматских начина слања података. Како би се користио овакав начин слања

података потребно је подесити тајмер који представља временски период након кога се врши слање пристиглих података. Практично након истека тајмера обавља се слање свих пристиглих података. Процес слања и пријема пакета података се обавља са потврдом пријема пакета која се шаље од стране сервера. Овај механизам је обезбеђен коришћењем могућности које пружа *TCP/IP* протокол.

Један од, такође, битних аспеката који се мора сагледати приликом одабира *GSM/GPRS* модула јесте и потрошња енергије. Техничком документацијом *SIM900* модула прописана потрошња енергије у моду мировања износи 1 mAh, док је у активном режиму рада потрошња 590 mAh. У моду мировања серијски порт престаје са радом што значи да је онемогућено слање *AT* команди. Током мода мировања са активном конекцијом, долазни саобраћај било са удаљеног сервера или клијента покреће акцију враћања модула у активни режим рада. Крајњи уређај, а који је у моду мировања, може се вратити у активни мод и приликом изостанка саобраћаја. Један од начина јесте постављање *DRT* (енг. *Data Terminal Ready)* пина на ниски сигнал у трајању од 20 ms. Други начин јесте коришћење *RTC* аларма. Коришћење *RTC* аларма се заснива на дефинисању времена трајања мода мировања, при чему ће након истека дефинисаног времена *RTC* аларма бити покренут механизам постављања уређаја у активни режим. Време успоставе конекције износи између 10-60 секунди. Након овог времена, модем је спреман за слање и пријем података. Време успоставе конекције зависи од брзине регистровања *SIM* картице у оквиру *GSM* мреже. Коришћење два мода рада *SIM900* модула омогућава значајне уштеде у погледу потрошње. Реализација слања података у оквиру система моделом описане метеоролошке станице посматрана је управо кроз коришћење ова два мода. Како је слање података планирано четири пута у току једног сата, преостали део времена *SIM900* модул се налази у моду мировања. Такође, с обзиром да су периоде у којима је *SIM900* модул активан унапред дефинисане, најадекватније је имплементирати прелазак из мода мировања у активни мод помоћу *RTC* аларма. Оваква активација *SIM900* модула једноставно је остварива софтверски, па самим тим нема додатних очекиваних активности од стране корисника. Практично, након сваког очитавања података и њихове припреме за слање, врши се активација *SIM900* модула, успостављање конекције и слање пакета податка на одредиште.

Економским издацима набавке *SIM900* модулао треба додати и цену *USB* адаптера. За потребе реализације повезивања *SIM900* модула са *Raspberry Pi* рачунарем, може се примети истоветни *USB* адаптер, као што је био случај приликом провезивања *GPS* модула. Такође, у зависности од провајдера, у цену реализације мора се укључити и месечни износ накнаде за пренос података *GPRS* мрежом. Међутим, с обзиром да се ради о малој количини података, као и с обзиром на чињеницу да основни тарифни пакети за минималну цену нуде сасвим довољну могућност преноса података, овај издатак свакако не утиче превише на цену реализације.

6.3 Поређење перформанси одабраног RF модула и GSM/GPRS модула

Комуникација моделом описане метеоролошке станице и базне станице коришћењем бежичних комуникационих система једна је од основних функционалности која мора бити имплементирана у циљу адекватног мониторинга и прикупљања метеоролошких параметара. Коришћење бежичних комуникационих система за потребе умрежавања, комуникације и преноса података пружа велике могућности у домену мобилности метеоролошке станице и аутоматизације њеног рада. Целокупан процес прикупљања метеоролошких и просторно-временских параметара коришћењем описаних сензора и *Raspberry Pi* рачунара у ни једном погледу не може се сматрати аутоматским уколико се измерене вредности ових параметара морају очитати на самој метеоролошкој станици. Имплементирање пакетног преноса података коришћењем бежичних комуникационих система својеврстан је и адекватан вид аутоматизације целог процеса. Могућа реализација преноса пакета података сачињених од вредности метеоролошких и просторно-временских параметара, анализирана је кроз реализацију бежичне мреже засноване на *RF* модулима и *GSM/GPRS* модулима. Анализирани радио фреквентни модули, као и *GSM/GPRS* модули, детаљно су описани у оквиру претходних поглавља. Поменута анализа базирана је на перформансама бежичног телекомуникационог система заснованог на коришћењу поменутих техника реализације комуникације и на могућностима остваривања пакетног преноса података.

Одабрани модули, како у једном тако и у другом случају, анализирани су на основу техничких карактеристика. Како је између различитих *RF* модула на основу захтеваних перформанси и ограничења у погледу коришћења на територији Европе одабран *ZigBee XBee S2C* 802.15.4 модул, перформансе овог *RF* модула су поређене са *SIM900 GSM/GPRS* модулом који је одабран као модул за реализацију бежичног телекомуникационог система коришћењем *GSM/GPRS* мреже.

Како би се обавило поређење перформанси два поменута комуникациона модула унапред су дефинисани критеријуми поређења који су у директној вези са потребама реализације комуникације метеоролошке станице са базном станицом. Параметри који су узети у обзир приликом поређења јесу: начин повезивања модула, потрошња електричне енегије, максимална могућа раздаљина између метеоролошке станице и базне станице у погледу преноса сигнала и брзина преноса пакета података.

Уколико се посматра остваривање основних техничких услова за креирање бежичног телекомуникационог система коришћењем било *RF* модула или *GSM/GPRS* модула на предајној и пријемној страни потребно је остварити услове за повезивање ових модула са одговарајућим рачунарем. Практично, на предајној страни *ZigBee XBee* или *SIM900* модул се повезује са *Raspberry Pi* рачунарем, док се на одредишној страни ови модули повезују са рачунарем који представља базну станицу. Најадекватнији начин повезивања ових модула са рачунарем, било на предајној било на пријемној страни, јесте коришћењем *USB* адаптер плоче. Сваки од модула користи адекватну *USB* адаптер плочу у зависности од распореда *GPIO* пинова, као и захтеваног начина реализације напајања и преноса података. Коришћење *Uni4 XBee/ZigBee* адаптер плоче са *USB* интерфејсом за потребе повезивања одабраног *RF* модула са рачунарем, као и *USB/TTL* адаптер плоче за повезивање *SIM900 GSM/GPRS* модула са рачунарем, изједначава ова два модула у домену повезивања. Међутим, свака од *USB* адаптер плоча карактерише се различом потрошњом електричне енерије. Посматрано из угла потрошње као једног од битних аспеката реализације, посебно на предајној страни услед коришћења соларног напајања метеоролошке станице, одабрани *RF* модуо заједно

са *USB* адаптер плочом користи укупно 545 mAh електричне енергије, док *GSM/GPRS* модул заједно са *USB* адаптером користи 590 mAh електичне енергије. Код оба модула је могуће остварити умањење потрошње преласком у мод мировања између свака два радна мода. Оба модула троше занемарљиво мало енергије у моду мировања, тачније максимално до 1 mAh. Такође, оба модула поседују могућност цикличног (у случају *ZigBee XBee* модула) или временског (у случају *SIM900 GSM/GPRS* модула) мода мировања. Овако реализовани модови мировања су потпуно истоветних карактеристика које се односе на повратак у активни режим рада након истека унапред дефинисаног кванта времена. Пренесено на уторшак енергије, потрошња *ZigBee XBee* 802.15.4 модула у току једног сата износи 36,33 mAh, док потрошња *SIM900 GSM/GPRS* модула током једног сата износи 39,33 mAh. Овако дефинисана потрошња оба модула одговара времену које ови модули проведу у активном моду рада током једног часа времена. Под временом проведеним у активном моду рада посматра се укупно време од почетка активације модула, преко успостављања везе и регистровања модула на мрежи, па све до успешног завршетка слања пакета података. Преостали период времена у току једног сата модули проводе у моду мировања, при чему се утрошак енергије може посматрати као истоветан, па самим тим и не фигурира у анализи ова два модула. С обзиром на истоветну потрошњу у моду мировања, ова потрошња не утиче на доношење одлуке о одабиру једног од ова два модула. Разлика у потрошњи између ова два модула износи 3 mA у корист *ZigBee XBee* 802.15.4 *RF* модула.

Поред начина повезивања и утрошка енергије, још један од параметара који је узет у обзир приликом упоредне анализе одабраног *RF* и *GSM/GPRS* модула јесте и максимална могућа удаљеност између метеоролошке станице и базне станице. Како се приликом реализације бежичног телекомуникационог система заснованог на употреби радио фреквентног модула јавља ограничење у погледу максималне пропагације сигнала у условима линије видљивости, раздаљина између метеоролошке станице и базне станице може представљати ограничавајући фактор. Максимална пропагација сигнала одабраног *ZigBee XBee S2C* 802.15.4 RF модула према техничким карактеристикама износи 1200 m. Остваривост домета од прописаних 1200 m, пре свега, условљена је линијом видљивости предајне и

пријемне антене. Такође, било какве препреке које могу утицати на директну линију видљивости утичу на пропагацију сигнала. Дирекна видљивост предајне и пријемне антене, с друге стране, ограничава могућност заштите овог модула од утицаја спољашњих фактора, пре свега климатских услова. Коришћење одабраног RF модула може се применити у случајевима када је 1200 m максимална раздаљина између метеоролошке и базне станице . Како се ради о производним површинама, постоји могућност појаве различитих препрека које могу утицати на пренос сигнала, па би самим тим и удаљеност моралабити мања. Уколико је удаљеност већа од 1200 метара, реализација бежичног телекомуникационог система заснованог на употреби овог модула изискује увођење нових хардверских уређаја у виду појачивача снаге сигнала.

Практично, на максималној удаљености од метеоролошке станице потребно је поставити појачивач који ће примљени ослабљени сигнал, добијен са *RF* предајника, проследити даље до одредишта или до новог појачивача у случајевима када раздаљина или конфигурација терена не могу бити савладане коришћењем једног појачивача снаге сигнала. Како би се на конкретној локацији могли поставити појачивачи снаге сигнала потребно је за исте обезбедити извор напајања. С тим у вези, њихово коришћење може узроковати набавку додатне опреме за потребе напајања што посредно повећава трошкове реализације бежичног телекомуникационог система, а самим тим и целокупног система метеоролошке станице. Коришћење *SIM900 GSM/GPRS* модула, с обзиром на карактеристике *GSM/GPRS* мреже, нема дефинисаних ограничења у погледу пропагације сигнала и максималне удаљености метеоролошке и базне станице. Потребан и довољан услов реализације бежичног телекомуникационог система коришћењем оваких модула на предајној и пријемној страни јесте покривеност обеју локација *GSM/GPRS* мрежом.

Коришћењем погодности које нуди ова бежична телекомуникациона мрежа пакетни пренос података се може обављати без увођења додатног хардвера, што није био случај код *ZiGBee* модула. Такође, *SIM900 GSM/GPRS* модул може адекватно радити и у затвореном простору, што оставља могућност његове заштите од спољашњих утицаја, посебно на страни метеоролошке станице.

Последњи у низу дефинисаних критеријума на основу којих је вршено поређење одабраног *RF* модула и *GSM/GPRS* модула јесте брзина преноса података. За *ZigBee XBee* фамилију радио фреквентних модула је карактеристично да могу остварити брзину преноса податка од 250 kb/s. Такође, коришћењем серијског преноса података код неких од фамилија *ZigBee XBee* радио фреквентних модула може се остварити брзина преноса од максимално 1 Mb/s. Уколико се посматра брзина преноса података код *SIM900 GSM/GPRS* модула, она иницијално, након првог покретања овог модула и успоставе *GPRS* мреже, износи 9600 kb/s. Подешавањем самог модула унапред дефинисаним *AT* командама може се остварити брзина преноса података од 115200 kb/s. Стављањем у однос брзине преноса података код једног и другог модула видљиво је да SIM900 GSM/GPRS модул остварује 38 пута већу брзину преноса података у основном моду рада. Основни мод рада *SIM900 GSM/GPRS* модула, такође, пружа приближно 9 пута већу брзину преноса података у односу на брзину преноса података у случају серијског преноса код *ZigBee XBee* фамилије радио фреквентних модула. Поређење брзине преноса података једног и другог модула показује *да SIM900 GSM/GPRS* модул има несумљиво већу брзину преноса података. Поред саме брзине преноса података, организација мреже применом *TCP/IP* модела и креирањем клијент сервер комуникације омогућава сигурнији процес преноса података. Један од параметара сигурности у оквиру преноса података свакако је механизам ретрансмисије обезбеђен од стране *TCP* протокола.

Целокупно поређење *ZigBee XBee S2C* 802.15.4 радио фреквентног модула са *SIM900 GSM/GPRS* модулом, базирано на набројаним параметрима, показало је да је *SIM900 GSM/GPRS* модул боље решење за реализацију комуникације у систему метеоролошка станица базна станица. Бежични телекомуникациони систем, заснован на коришћењу овог модула, омогућава већу мобилност саме метеоролошке станице, независност комуникације од међусобне удаљености метеоролошке и базне станице, као и већу брзину преноса података. Већа потрошња енергије овог модула у односу са одабрани RF модул је занемарљива чињеница наспрам свих наведених предности.

7. Анализа креираног скупа података

Примена data mining предикционих техника захтева употребу скупа података са познатим исходом. За успешно креирање шаблона над подацима, као и за успешно дефинисање законитости између података, потребно је креирати или прибавити одговарајући скуп података над којим ће се примењивати одговарајући предикциони алгоритми. Предикција времена хемијских третмана у склопу овог истраживања је заснована на скупу података сачињеном од вредности метеоролошких и просторно-временских параметара. Како би се добио скуп података са познатим исходом извршено је спајање креираног скупа метеоролошких и просторно-временских података са скупом података који представља информацију о појави конкретно посматраних болести. Како се ради о скупу података са познатим исходом, управо појава болести представља завистан атрибут чије су вредности у оквиру тренинг скупа података познате. У креираном скупу података вредности метеоролошких и просторно-временских параметара представљају независне атрибуте.

Прикупљене вредности метеоролошких и просторно-временских података, као и података о појави конкретних болести, датирају из прошлости. Поред чињенице да дате вредности датирају из прећашњег времена, незаобилазна је чињеница да је скуп података креиран спајањем два скупа података. Управо из ових разлога, потребно је прикупљене податке организовати у јединствену базу података. Креирање јединствене и по структури најједноставније базе података омогућава лакшу анализу података у оквиру саме базе као и коришћење истих. Анализа података, пре свега, огледа се у могућностима редукције базе податка у циљу смањења димензионалности базе података. Такође, поред редукције иницијално креирне базе података извршена је анализа у погледу детекције недостајућих вредности атрибута, детекције *outlier*-а и отклањања свих уочених недостатака. Решавање евидентираних недостатака у оквиру базе података извршено је применом неких од описаних метода за решавање оваквих проблема. С обзиром на евидентирани недостатак, одабрана је најефикаснија метода. На пример, за потребе детекције *outlier*-а примењен је већи број метода.

Применом већег броја метода очекивано је детектовање већег броја *outlier*-а, као и њихово ефикасно решавање. Такође, након детекције и реализације недостајућих вредности атрибута применом предикције, као једног од могућих метода одређивања недостајућих вредности атрибута, примењено је поновно детектовање *outlier*-а. Поновна детекција *outlier*-а извршена је како би се потврдило да предикцијом добијена вредност припада опсегу вредности за дати атрибут.

Потребно је нагласити да су процес детекције недостајућих вредности атрибутаи процес детекције и отклањања *outlier*-а оријентисани ка креирању базе података са најтачнијим могућим скупом података. Применом предикционих data mining алгоритама над скупом података у оквиру кога су реализовани сви евидентирани недостаци, може се очекивати креирање адекваног предикционог модела. На основу свих наведених чињеница, може се закључити да је тачност предикције директно условљена тачношћу података у оквиру скупа података са познатим исходом. Наравно, поред тачности скупа података, кључну улогу заузима и одабир адекватног предикционог алгоритма, као и евалуација креираних предикционих модела са циљем одабира најадекватнијег.

7.1 Организација прикупљених података и креирање базе података

Креирани скуп података обухвата вредности метеоролошких и просторно-временских параметара за период од 2009. до 2016. године. Креирани скуп података је базиран на метеоролошким извештајима јавно публикованим од стране Републичког хидрометеоролошког завода Републике Србије. Преузети извештаји о стању метеоролошких услова за поменути период су организовани у оквиру једноставне базе податка. База података за потребе анализе података и креирања предикционих модела, као и одабира најбољег предикционог алгоритма, креирана је у оквиру *.xlsx* фајла. Једноставност базе података огледа се у табеларном приказу података, при чему се фајл базе података састоји од само једне табеле. На овакав начин је онемогућено преклапање табела са подацима, као и мешање података, јер су сви подаци организовани у оквиру једне табеле.

Инстанце података су у оквиру базе података организоване на нивоу редова, што значи да су вредности независних и зависних атрибута организоване на нивоу колона. Метеоролошки подаци представљају првих шест колона у оквиру креиране табеле са подацима. Ових шест колона одговара минималној температури ваздуха, максималној температури ваздуха, средњој дневној температури ваздуха, просечној влажности ваздуха, количини падавина и брзини ветра. Вредности метеоролошких параметара су добијене на следећи начин: минимална температура је добијена као минимална вредност свих мерења температуре ваздуха у току једног дана, максимална температура је добијена као максимална вредност свих мерења температуре ваздуха у току једног дана, просечне вредности температуре ваздуха, влажности ваздуха и брзине ветра су добијене изачунавањем просечне вредности на основу целодневних мерења ових параметара. Количина падавина је добијена као сума свих регистрованих падавина током дана. Практично, на овакав начин свака инстанца података у оквиру базе података представља један дан.

Просторно-временски параметри у оквиру базе података дефинисани су редним бројем месеца коме одговара дан у коме су мерења вршена. Редуковање скупа просторно-временских параметара на ознаку месеца, у коме је дато мерење вршено, биће детаљније описано у следећем поглављу. Последња колона скупа података представља вредност класног арибута, а који представља предефинисану вредност која указује на присуство посматране болести. Како је истраживање због проверљивости резултата засновано на предикцији времена хемијских третмана две познате болести воћа, предефинисане вредности класних атрибута су: *monilia*, *coccomyces*, *both* и *nothing*. Прва вредност указује да су остварени услови за појаву *monilinie*, друга вредност указује на оствареност услова за појаву *coccomyces*, док трећа и четврта вредност указују на то да су остварени услови за појаву обе болести или да нема потребних услова за појаву било које од две поменуте болести респективно. Овако креирана база података је креирана из целокупног скупа података који је обухватао додатне метеоролошке и просторно-временске параметре. Скуп метеоролошких и просторно-временских параметара је редукован на поменути скуп података избацивањем оних атрибута чије вредности нису од утицаја на саму предикцију.

Редуковане вредности могу бити од значаја у процесу предикције остварености услова за појаву неких других биљних патогена или штеточина, те је значајно њихово чување у оквиру иницијално креираног скупа података.

Креирана база података се може сматрати иницијалом базом података са свим потребним параметрима за креирање предикционог модела. Практично, на основу креиране иницијалне базе података креирају се две засебне базе података које се каније користе у процесу обуке и тестирања предикционог модела. Прва од креираних база података представља тренинг скуп података на основу кога се врши обука предикционог модела, док друга база података представља тест скуп података на основу кога се врши евалуација предикционих модела. Над иницијално креираном базом података са циљем добијања адекватнијег скупа података извршена је редукција података. Редукција података је базирана на чињеници да током периода мировања биљке од јануара до марта, закључно са мартом месецом, не постоје услови за развој било које од посматраних болести. С тим у вези, могуће је редуковати скуп података избацивањем свих инстанци података које представљају мерења метеоролошких и просторно-временских параметара током поменутог периода. Такође, период активне вегетације коштичавог воћа, које је узето као предмет истраживања, траје закључно са августом месецом, те је самим тим могуће избацити из базе података инстанце података које одговарају мерењима начињеним од септембра до децембра месеца. На овакав начин, у оквиру иницијално креиране базе података након редукције остају инстанце података које представљају период од априла до августа.

Пренесено на посматрани период од 2009. до 2016. године укупан број инстанци у бази података износи 1224. Овако креирана база података се може посматрати као тренинг скуп података. Како се ради о бази података са познатим исходом, за потребе будуће евалуације предикционих модела, креирана је још једна база података која представља тест скуп података. У оквиру базе података која представља тест скуп података случајним избором из базе података која представља тренинг скуп, издвојено је 224 инстанци података. Овако креиран тест скуп података омогућава правилну евалуацију предикционох модела, креираних на основу зависности између података у оквиру тренинг скупа.

Како инстанце подака које представљају тест скуп не би биле коришћене у процесу креирања модела, исте су избачене из тренинг скупа података, што овај скуп редукује на 1000 инстанци. Овако креирани тренинг и тест скуп података представљају основу за креирање прототипа апликације у оквиру које је имплементирано креирање предикционих модела и њихова евалуација са циљем одабира најадекватнијег.

7.2 Анализа података и отклањање *outlier*-а

Како би се креирали адекватни скупови података који представљају тренинг и тест скупове података, анализа самих података и детекција *outlier*-а извршена је над иницијално креираном базом података. Тачније, целокупна анализа података је извршена над базом података која је настала након редукције података. Практично поменута база података са укупно 1224 инстанце података представља полазни скуп података над којим је потребно извршити анализу и из истог отклонити све *outlier*-е. Посматрањем структуре података у оквиру базе података уочљиво је да су сви метеоролошки и просторно-временски подаци изражени нумерички, што омогућава њихово статистичко обрађивање. Класни атрибут, са друге стране, с обзиром да представља информацију о појави болести, изражен је номинално. Унапред дефинисане номиналне вредности класног атрибута описане су раније.

Статистична анализа нумеричких вредности атрибута у оквиру креиране базе података дата је у табели 13. Спроведеном статистичком анализом за сваки од нумеричких атрибута израчуната је минимална, максимална и средња вредности. Такође, поред ових вредности извршено је израчунавање варијансе и стандардне девијације. Средња вредност нумеричког атрибута и стандардна девијација представљају параметре потребне за детекцију могућих *outlier*-а. Метод детекције *outlier*-а у коме се користе средња вредност и стандардна девијација, а који је заснован на Чебишевљевој теореми описан је раније. Како се ради о методу детекције који захтева израчунавање одступања конкретне вредности атрибута у оквиру инстанце података од стандардне девијације целог скупа података, потребно је дато израчунавање извести за сваку инстанцу података понаособ.

Табела 13: Статистичка анализа нумеричких вредности атрибута

Назив атрибута	Минимална вредност	Максимална вредност	Просечна вредност	Варијанса	Стандарна девијација
Мин. температура ваздуха [ºC]	-2	28	12,96	24,52	4,95
Мах. температура ваздуха [ºC]	5	40	26,26	47,14	6,87
Просечна температура ваздуха [ºC]	3	33	19,60	29,93	5,47
Просечна влажност ваздуха [%]	0,31	0,97	0,63	0,02	0,13
Количина падавина [mm]	0	99	1,34	41,66	6,45
Брзина ветра [km/h]	3	27	7,59	16,63	4,08

Како би се омогућило детектовање потенцијалних *outlier*-а на основу дефинисаних вредности корелационог фактора у склопу креиране базе података, као и у будућим базама података насталим изменом и допуном постојеће, извршена је имплементација овог метода. Имплементација дефинисаног метода детекције потенцијалних *outlier*-а доноси предности у погледу брзине извршења детекције и независности од одабране вредности корелационог фактора и броја инстанци података у бази података. Овако имплементирани поступак детекције потенцијалних *outlier*-а поновљен је укупно 24 пута. Практично, потенцијални *outlier*-и детектују се посебно за сваки од метеоролошких параметара, што укупно чини шест пролаза кроз базу података. Такође, за сваки од метеоролошких атрибута поступак детекције је потребно поновити са сваком од четири дефинисаних вредности корелационог фактора. Промена вредности корелационог фактора се користи како би се одредио корелациони фактор чијом се применом добија најмањи број лажно позитивних или лажно негативних резултата детекције потенцијалних *outlier*-а.

Резултати извршења детекције потенцијалних *outlier*-а су приказани у табели 14. Анализа добијених резултата показује да се повећањем корелационог коефицијента смањује број детектованих потенцијалних *outlier*-а.

Табела 14: Број детектованих outlier-*а за различите вредности корелационог коефицијента ε*

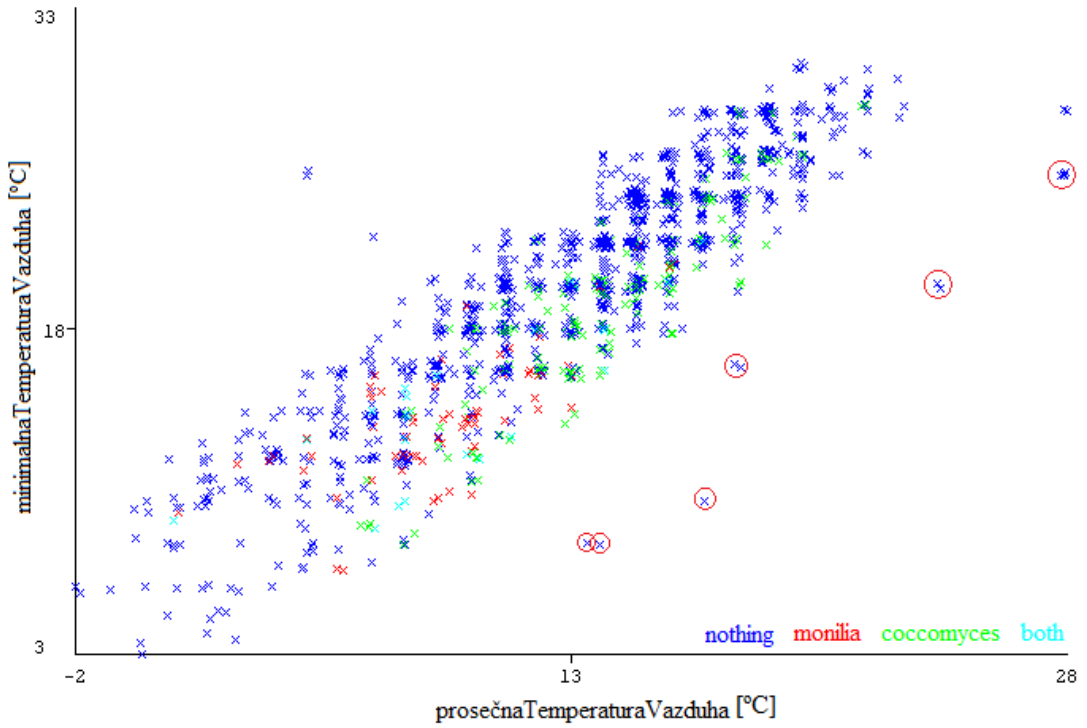
	Вредности корекционог коефицијента			
Назив атрибута	$\varepsilon = 3$	$\varepsilon = 4$	$\varepsilon = 5$	$\varepsilon = 6$
Мин. температура ваздуха	6	0	0	0
Мах. температура ваздуха	0	0	0	0
Просечна температура ваздуха	0	0	0	0
Просечна влажност ваздуха	0	0	0	0
Количина падавина	10	6	4	4
Брзина ветра	27	8	0	0

Након извршења имплементираног решења за детекцију потенцијалних *outlier*-а, добијени резултати показују различито присуство потенцијалних *outlier*-а за различите вредности корекционог коефицијента. Највеће присуство *outlier*-а је регистровано за атрибут који представља брзину ветра, и то за најмању вредност корекционог коефицијента. Присуство потенцијалних *outlier*-а за атрибут који представља количину падавина регистровано је за сваку од вредности корекционог коефицијента, па се самим тим може претпоставити да овај атрибут заиста и садржи вредности које представљају *outlier*-е. Како се сви детектовани *outlier*-и посматрају као потенцијални, потребно је извршити додатну проверу са циљем одбацивања лажно детектованих потенцијалних *outlier*-а. Како је детекција потенцијалних *outlier*-а вршена за различите вредности корелационог коефицијента извршен је пресек инстанци података у којима се налази потенцијални *outlier*. На овакав начин се креира скуп потенцијалних *outlier*-а у који су уврштени само они *outlier*-и који се појављују у израчунавањима за сваку од вредности корелационог коефицијента. Овако креирани скуп детектованих потенцијалних *outlier*-а садржи четири инстанце података које се појављују у свакој од покренутих детекција *outlier*-а за количину падавина. Детектовани потенцијални *outlier*-и за количину падавина представљају инстанце у којима се количина падавина појављује са максималном количином падавина од 99 mm/m^2. Како дата вредност представља велико одступање од просечне вредности за

количину падавина, предметне инстанце су означене као потенцијални *outlier*-и. Прегледавањем базе података је установљено да се ради о инстанцама под редним бројем 525, 526, 832 и 833. Поређењем креиране базе података са основним скупом метеоролошких и просторно-временских података, утврђено је да вредност за количину падавина од 99 mm/m^2 у инстанцама 832 и 833 не представља *outlier*. Поменуте инстанце садрже метеоролошке и просторно-временске податке који представљају дане током јуна месеца 2014. године, када је на поменутом региону долазило до појаве циклона са надпросечно великом количином падавина. Вредности за количину падавина у оквиру инстанци 525 и 526 заиста представљају *outlier*-е. Ови *outlier*-и су настали погрешним уносом оригиналних вредности за количину падавина за дане које интанце података представљају. Поменути *outlier*-и су откоњени уносом коректних вредности за дате дане.

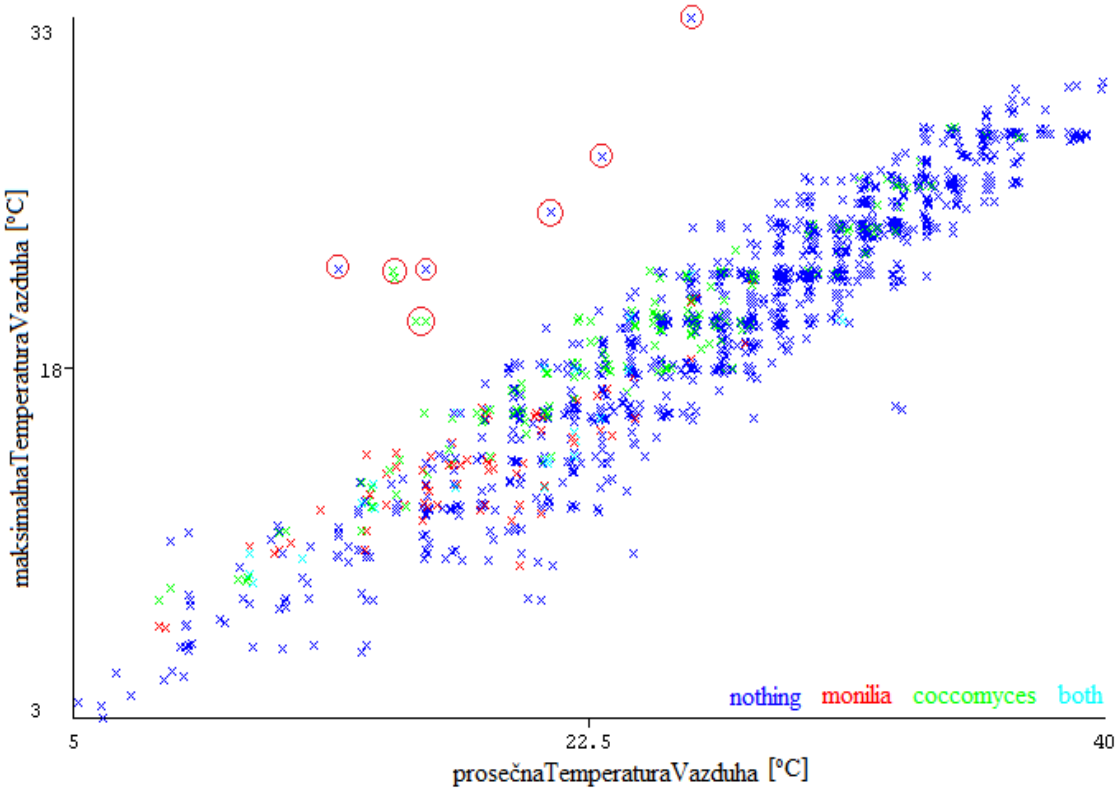
Креирањем пресека скупова детектованих потенцијалних *outlier*-а, као и поређењем вредности које су означене као *outlier*-и са стварним вредностима у оригиналном скупу података, утврђено је да су детектовани потенцијални *outlier*-и за минималну температуру и брзину ветра лажно детектовани. На основу свега наведеног, може се закључити да примена овакве методе даје најбоље резултате када се ради са корелационим фактором $\varepsilon = 5$. Овако дефинисана метода детекције *outlier*-а, заснована на статистичким параметрима скупа податка, најчешће је праћена визуелизационим методама. Управо из ових разлога је дати скуп података визуелизован по сваком од параметра. Стављањем у однос различитих параметара добија се могућност уочавања аномалија у скупу података које могу представљати потенцијалне *outlier*-е. За потребе визуелизације параметара у скупу података коришћен је *WEKA* алат и *MatLab* развојно окружење. Практично, извршена је визуелизација скупа података по сваком од метеоролошких параметара. Визуелизацијом се добија просторна расподела вредности сваког од атрибута који представљају метеоролошке параметре, што омогућава уочавање инстанци података које одступају од креиране расподеле. Најпре је креирана расподела минималне и максималне температуре ваздуха у односу на средњу дневну температуру ваздуха. Овако креираном визуелизацијом детектовани су *outlier*-и који указују на аномалије између минималне температуре

ваздуха и средње дневне температуре ваздуха, као што се може видети на слици 43. Са друге стране, на слици 44 означени су *outlier*-и који представљају аномалије детектоване стављањем у однос максималне температуре ваздуха и средње дневне температуре ваздуха. Детектовани *outlier*-и су означени црвеним кружићем око ознаке за конкретну инстанцу података. У оквиру инстанци података које су применом визуелизације детектоване, *outlier*-и се огледају у томе да је средња дневна температура мања од минималне или већа од максималне температуре ваздуха. *Outlier*-и детектовани на овакав начин, са једне стране могу указивати на грешку у вредностима за минималну или максималну дневну температуру ваздуха, док са друге стране грешка може бити у оквиру вредности за средњу дневну температуру ваздуха. Овако детектовани и означени *outlier*-и се лако лоцирају у оквиру базе података, с обзиром да се са визуелног приказа одабиром конкретног *outlier*-а добијају све информације које носи инстаца података у оквиру које се налази детектована *outlier* вредност. Поређењем вредности у оквиру базе података са изворним вредностима у оквиру иницијалног скупа података извршено је отклањање детектованих *outlier*-а.

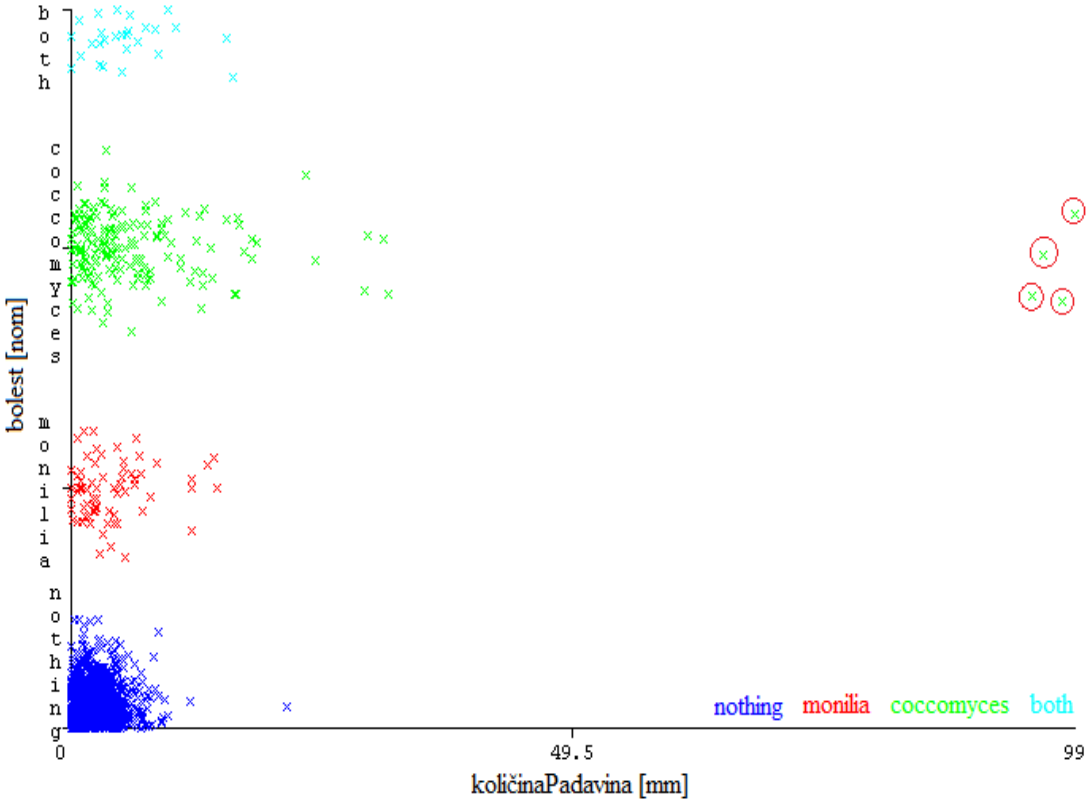


Слика 43: Визуелизација минималне температуре ваздуха у односу на средњу дневну температуру ваздуха

Потврда претходно детектованих и потврђених *outlier*-а за количину падавина још једном се добија визуелизацијом података за количину падавина. Подаци за количину падавина се могу ставити у однос са вредностима класног атрибута или са вредностима било ког другог атрибута који представља метеоролошке параметре. Визуелизација количине падавина је приказана на слици 45. Детектована четири потенцијална *outlier*-а применом статистичке методе се јасно издвајају и приликом визуелизације. Како се ради о визуелизацији података без унетих измена у оквиру инстанци података које предсављају *outlier*-е за количину падавина, очекивана је детекција означених вредности. Поред потврде потенцијалних *outlier*-а детектованих статистичким методама, потребно је нагласити и појаву лажне детекције за исте инстанце података, што је био случај са претходно примењеним статистичким методама над атрибутом за количину падавина.



Слика 44: Визуелизација максималне температуре ваздуха у односу на средњу дневну температуру ваздуха



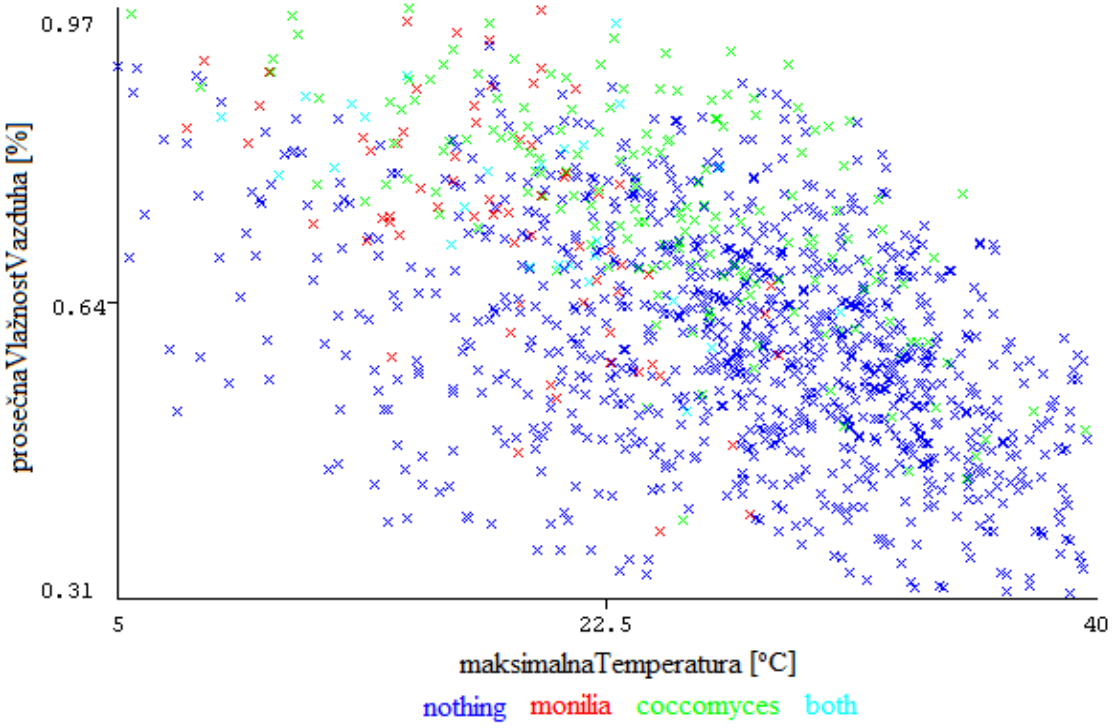
Слика 45: Визуелизација количине падавина у односу на болести

Овако извршена потврда детектованих *outlier*-а несумљиво показује да се ради о вредностима које заиста представљају *outlier*-е.

Последња два атрибута у оквиру креиране базе података са познатим исходом који представљају метеоролошке параметре, а за које је вршена визуелизација са циљем детекције *outlier*-а, јесу релативна влажност ваздуха и брзина ветра. Претходно спроведена статистичка анализа за реалитивну влажност ваздуха, као што се може видети у табели 14, није показала постојање *outlier*-а, док је применом мање вредности корекционог фактора (вредности $\varepsilon = 3$ и $\varepsilon = 4$) статистичка анализа детектовала 27, односно 8 потенцијалних *outlier*-а респективно.

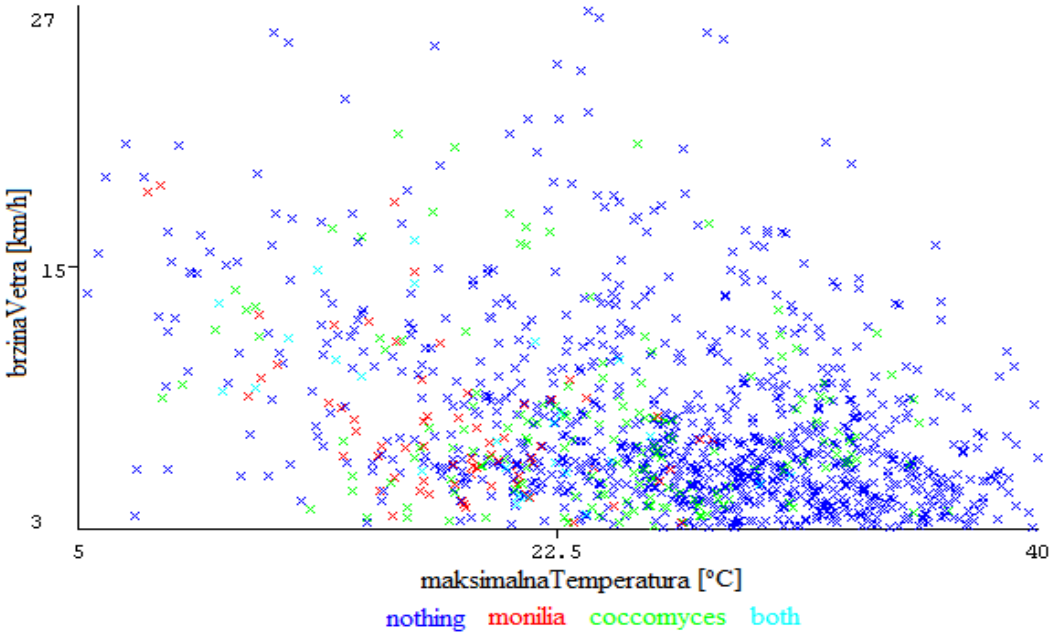
Како је на примеру за количину падавина поређењем и визуелизацијом потврђено да статистичка анализа даје најбоље резултате за вредност корекционог фактора $\varepsilon = 5$, могло би се сматрати да детектовани потенцијални *outlier*-и за брзину ветра представљају лажне резултате.

Како је познато да се резултати детекције потенцијалних *outlier*-а поменутом статистичком методом најчешће потврђују визуелизацијом, извршена је визуелизација атрибута који представљају релативну влажност ваздуха и брзину ветра. Креирана визуелизација није показала присуство инстанци података које би се могле окарактерисати као потенцијални *outlier*-и, што се може видети на слици 46 и слици 47, а за релативну влажност ваздуха и брзину ветра респективно.



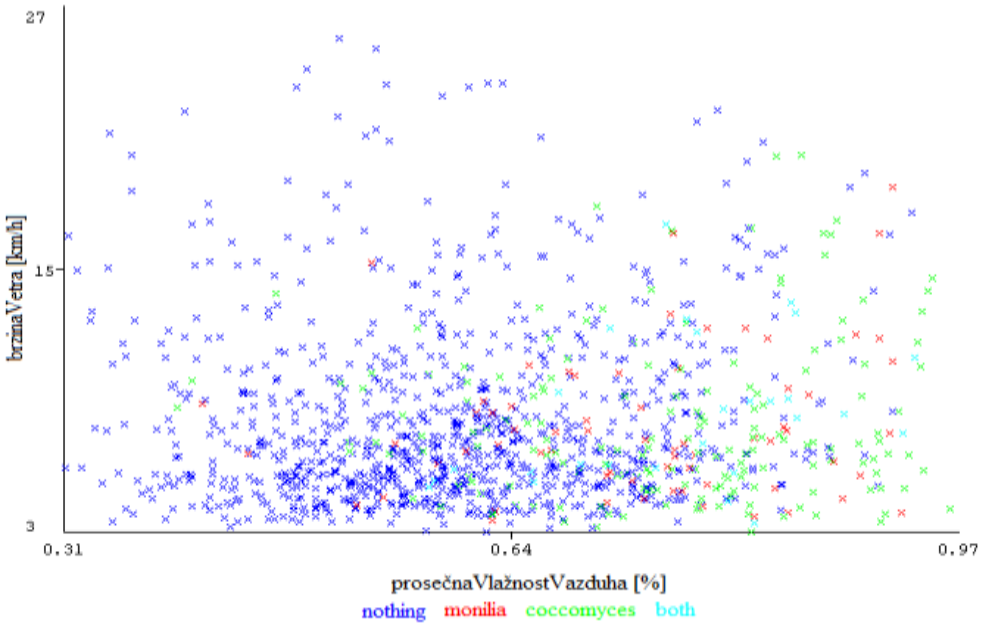
Слика 46: Визуализација релативне влажности ваздуха у односу на максималну температуру

Како се може видети са слике 46 и слике 47, визуелизација релативне влажности ваздуха и брзине ветра креирана је стављањем вредности ових параметра у однос са вредностима за максималну дневну температуру ваздуха. Потребно је нагласити да су поред приказаних визуелизација креиране и визуелизације стављањем у однос предметних атрибута (релативна влажност ваздуха и бризина ветра) са осталим атрибутима у бази података. Овако креиране визуелизације, такође, нису показале присуство *outlier*-а па из тог разлога визуелизациони дијаграми нису приказани.



Слика 47: Визуелизација брзине ветра у односу на максималну температуру

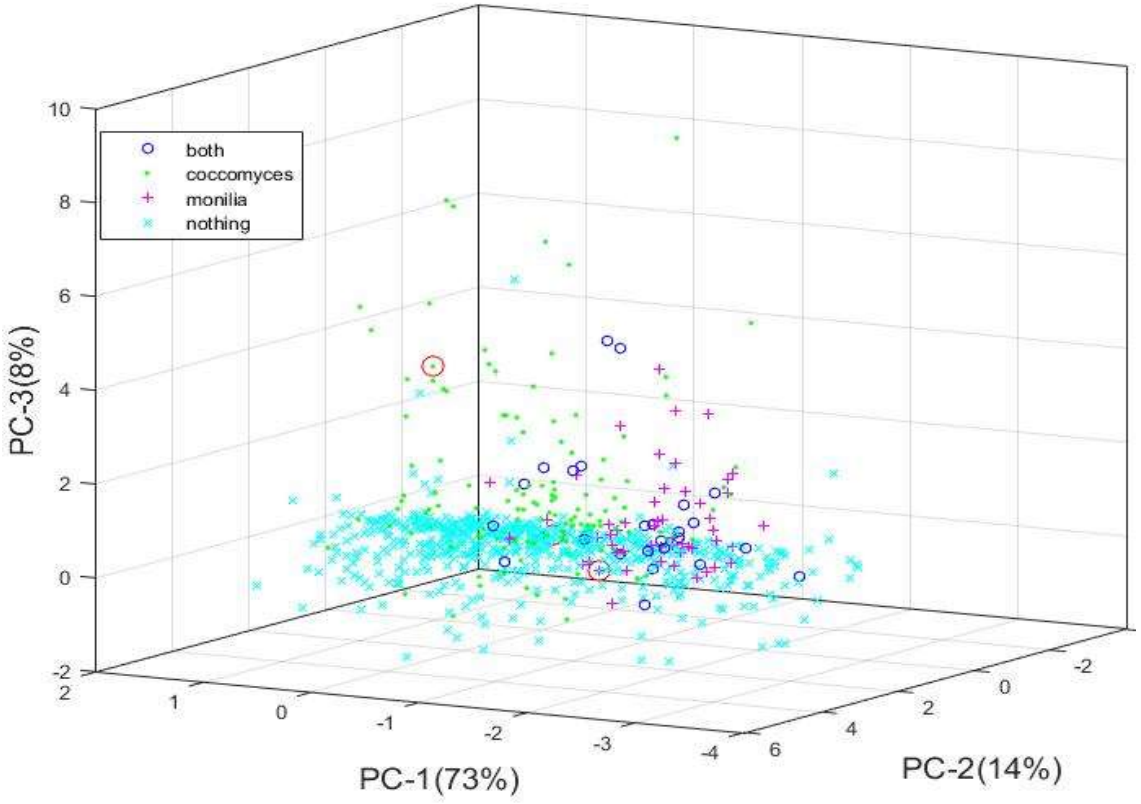
Последња у низу визуелизација, креирана са циљем детекције потенцијалних *outlier*-а у оквиру вредности за релативну влажност ваздуха и брзину ветра, јесте визуелизација заснована на међусобном односу вредности ова два атрибута. Визуелизација односа вредности ова два атрибута дата је на слици 48.



Слика 48: Визуелизација односа између просечне влажности ваздуха и брзине ветра

Прегледом дијаграма визуелизације на слици 48, као и прегледом вредности за поменуте атрибуте на самом дијаграму, не уочавају се потенцијални *outlier*-и. Практично, на основу непостојања екстремних одступања било које од инстанци података од шаблона који прате остале инстанце података, може се закључити да атрибути који представљају релативну влажност ваздуха и брзину ветра у скупу својих вредности не садрже вредност која може бити потенцијални outlier.

Додатна статистичка анализа вредности ознаке месеца није потребна, с обзиром да нумеричка ознака представља само редни број месеца у години коме дата инстанца припада. Ознака месеца у оквиру базе података је уведена како би се отклонили потенцијални *outlier*-и који се односе на однос вредности независних и зависних аутрибута у оквиру инстанци података. рименом технике визуелизације инстанци података уочен је велики проблем преклапања вредности класних атрибута за различите инстанце података. Овакво поклапање је евидентирано између инстанце 333 и инстанце 646. Како би се обезбедила боља визуелизација података у бази података, па самим тим и односа вредности независних и зависних атрибута, имплементирана је *PCA* анализа. На основу принципа редукције димензионалности података на коме је заснована *PCA* анализа су одабране прве три *principal* компоненте као основне димензије за спровођење анализе и визуелизације података. Три поменуте компоненте су одабране јер имају највеће вредности. Слика 49 представља репрезентацију података у три димензије, а на основу спроведеног израчунавања *PCA* анализе. Поменуте *principal* компоненте означне су са PC-1, PC-2 и PC-3 респективно [51]. Тродимензионална репрезентација обезбеђује бољу визуелизацију података засновану на могућностима ротације. Овако спроведеном визуелизацијом се могу детектовати проблеми који се огледају у томе да се за исте вредности независних атрибута који представљају метеоролошке параметре у оквиру различитих инстанци података јављају различите вредности зависног атрибута. Пренесено на конкретан проблем, две инстанце података (333 и 646) имају следеће вредности метеоролошких параметара 10, 23, 16, 0.39, 0 и 10 које одговарају минималној, максималној, средњој дневној температури ваздуха, просечној влажности ваздуха, количини падавина и брзини ветра респективно.



Слика 49: Детекција outlier-*а визуелизацијом података у три димензије заснованом на PCA анализи*

Притом, номинална вредност класног атрибута инстанце 333 је *monilia*, док је номинална вредност класног атрибута инстанце 646 *nothing*. Посмарањем ових двеју инстанци података, са једне стране (инстанца 333) се долази до закључка да су дати метеоролошки услови повољни за остваривање инфекције *monilini*-ом, док се са друге стране (инстанца 646) долази до закључка да су дати метеоролошки услови неповољни за развој било које од двеју посматраних болести. Ситуација у којој се јављају две категорички различите вредности класног атрибута за исте вредности независних атрибута сигнализира на појаву *outlier*-а за неку од вредности, како незивисног тако и зависног атрибута. Прегледом и провером вредности ових двеју инстанци података установљено је да су вредности метеоролошких параметара тачне. Основна разлика између ових двеју инстанци подака је у ознаци месеца, што их чини обема исправним. Практично, исти метеоролошки услови остварени током априла месеца припадају инстанци 333, док инстанца 646 репрезентује поменуте метеоролошке параметре током августа месеца.

Посматрано са фитопатолошке стране април је месец током кога су споре *monilie* активне и током кога, уз одговарајуће метеоролошке услове, може доћи до инфекције. Са друге стране, уз остварење истоветних метеоролошких услова период инфекције посматраних двеју болести током августа месеца знатно је смањен, те класни атрибут *nothing* коректно указује да неће доћи до инфекције. Сличан проблем се може јавити у случајевима када метеоролошки параметри одговарају условима за инфекцију *coccomyces hiemalis*-ом. Уколико су одговарајући метеоролошки параметри измерени током априла месеца, када нема активности овог патогена, не може ни доћи до његовог развоја. Управо је ознака месеца у овом случају пресудан параметар на основу кога се може извршити успешна предикција вредности класног атрибута. Оваква чињеница оправдава употребу ознаке месеца као просторно-временског податка у оквиру тренинг и тест базе података, као и приликом креирања базе података за употребу од стране крајњег корисника.

Ознака месеца у домену анализе података у оквиру креиране базе података може се искористити за потребе одређивања броја мерења у току једног месеца. Провером броја појављивања ознаке месеца у оквиру базе података добија се информација о томе да ли број инстанци података у којима се појављује предметна ознака одговара броју дана који дати месец има. Уколико је број појављивања мањи од броја дана, евидентира се недостатак мерења, а самим тим и недостатак инстанце. Са друге стране, уколико је број појављивања већи од броја дана, евидентира се постојање инстанце која не одговара датом месецу. У случајевима већег броја инстанци података од дозвољеног за један месец потребно је утврдити да ли није дошло по дуплицирања неке од инстанци података за дати месец. Такође, у оваквом сличају, уколико укупан број инстанци одговара укупном броју дана за посматрани период, појава већег броја инстанци за један месец указује да ће се неки други месец јавити са мањим бројем мерења. Случајеви недостатка инстанце података, при чему је и укупан број инстанци података за посматрани период мањи од потребног, носе мање проблема од случајева у којима је потребно утврдити која инстанца посматраног скупа података је дуплицирана или унета као погрешна.

Решавање оваквог проблема се огледа у провери иницијалног скупа података који је коришћен за креирање базе података са познатим исходом.

Номиналне вредности класног атрибута, с обзиром на своју специфичност у оквиру креиране базе података, не могу се подвргнути посебној анализи са циљем детекције потенцијалних *outlier*-а. Тачније, примењена *PCA* анализа, као што је раније наведено, дала је ефекта у откривању инстанци у којима је могуће постојање непоклапања вредности независних атрибута и вредности класног (зависног) атрибута. Оваква анализа није пронашла аномалију у оквиру вредности класних атрибута, већ у оквиру корелације вредности независних и зависног атрибута, што је отклоњено додавањем ознаке месеца. Као једна од евентуалних анализа вредности класног атрибута спроведена је анализа у домену одређивања броја инстанци које имају као вредност класног атрибута неку од унапред дефинисаних номиналних вредности. Од укупног броја инстанци података у бази података, 957 инстанци података за вредност класног атрибута имају вредност *nothing*, 68 инстанци имају вредност *monilia*, 172 инстанце имају вредност *coccomyces*, док 27 инстанци имају вредност *both*. Оваква анализа је показала да свака од инстанци података садржи искључиво неку од предефинисаних вредности класног атрибута.

Спроведена стастистичка анализа, као и анализа базе података заснована на визуелизацији података, имале су за циљ детекцију оних вредности у оквиру базе података које се могу сматрати *outlier*-има. Од укупног броја инстанци у оквиру базе података применом описаних поступака детекције *outlier*-а детектовано је 27 инстанци података у оквиру којих за неки од атрибута постоји вредност супротна од очекиване. Сваки од детектованих и потврђених *outlier*-а успешно је реализован заменом *outlier* вредности одговарајућом вредношћу за дати атрибут. Визуелизација података, поред детекције *outlier*-а, указала је и на потребу увођења ознаке месеца као једног од атрибута у оквиру скупа података. Значајност увођења оваквог атрибута из угла анализе података није велика, међутим, из домена тачности предикционог модела итекако је значајна.

8. Креирање предикционог модела

Прикупљање метеоролошких и просторно-временских података, као података који представљају вредности независних атрибута, и њихово обједињавање са вредностима зависног класног атрибута у оквиру јединствене базе података представља припрему одговарајућих података који ће се користити у процесу креирања предикционог модела. Анализа поменутих података, оријентисана ка детекцији и отклањају *outlier*-а у креираном скупу података, неизоставан је процес који претходи процесу креирања предикционих модела. Након што су поменута два процеса (креирање и анализа базе података са познатим исходом) адекватно спроведена, могуће је покренути процес креирања предикционих модела и њихову евалуацију са циљем одабира најбољег модела.

Предикциони модели се могу креирати применом различитих математичких и data mining метода. Тачност процеса предикције непознатих вредности класних атрибута, на основу скупа података са познатим исходом, у основи је условљена примењеном математичком или data mining методом. Прецизније, тачност креираног предикционог модела је зависна од корелационог односа између независних и зависних атрибута, што значи да сваки од независних атрибута у процесу креирања предиктивне вредности класног атрибута учествује са различитим степеном значајности. Самим тим, прецизност предикционог модела се огледа у препознавању степена утицаја појединачног независног атрибута на коначну предикцијом добијену вредност зависног атрибута. Сваки од предикционих метода током процеса тренинга предикционог модела проналази поменуте везе између зависног атрибута и сваког од независних атрибута. Управо на овакав начин се врши креирање предикционих модела на основу којих се у процесу предикције врши одређивање вредности класног атрибута.

Постојање великог броја предикционих метода отвара могућност одабира оних који ће као резултат примене дати најбоље предикционе моделе за дати скуп податка. Како би се од већег броја математичких и data mining метода одабрале оне методе на основу којих је најбоље креирање предикционих модела за дати скуп података у овом истраживању је креиран прототип апликације намењене крајњим корисницима.

Задатак протипа апликације била је имплементација и прилагођавање познатих метода за примену над датим скупом података како би се помоћу њих могло извршити креирање предикционих модела. Такође, у оквиру имплементираног прототипа апликације, имплементирана је раније поменута функционалност визуелизације података, како са циљем детекције *outlier*-а у скупу података тако и са циљем креирања визуелног приказа међусобних веза између података. Креирани прототип апликације обухвата процес уноса података, обуку предикционог модела коришћењем тренинг скупа података и евалуацију обученог предикционог модела. Процес евалуације креираног и обученог предикционог модела је заснован на коришћењу унапред креираног тест скупа података у коме се налазе инстанце података са познатим исходом.

Имплементација поменутог прототипа апликације је извршена коришћењем *MatLab* програмског пакета и *Weka* data mining алата. *MatLab* је програмски пакет који се користи за решавање математичких проблема и израчунавања у различитим областима науке и технике. *MatLab* обухвата програмирање, израчунавање и визуелизацију. Низ или матрица је основна форма за податке, а која не захтева димензионисање. Назив *MatLab* потиче од *Matrix Laboratory*. *MatLab* садржи алате за решавање специфичних проблема. Ови додатни програмски пакети се називају *toolbox*. *Toolbox* представља колекцију *Matlab* функција које пружају могућност да се помоћу *MatLab*-а реше проблеми из различитих области: обрада сигнала, управљање процесима, телекомуникације, симулације и др. За потребе овог истраживања је у оквиру *MatLab* програмског пакета извршено додавање *Statistics and Machine Learning Toolbox-a*. Основне анализе, спроведене поступком статистичке обраде, као и поступком визуелизације, су извршене коришћењем алата и функционалности доступних у оквиру овог *toolbox-a. Weka*, као други поменути алат, представља бесплатан data mining алат. Заправо, *Weka* представља скуп data mining алата у оквиру јединственог окружења. Предност коришћења Weka алата је то што је врло лак за употребу. Дизајниран је у Java програмском језику и првенствено је био креиран у сврху истраживања на Универзитету Wakaito, али је касније постао глобално прихваћен. Користи се највише у домену машинског учења.

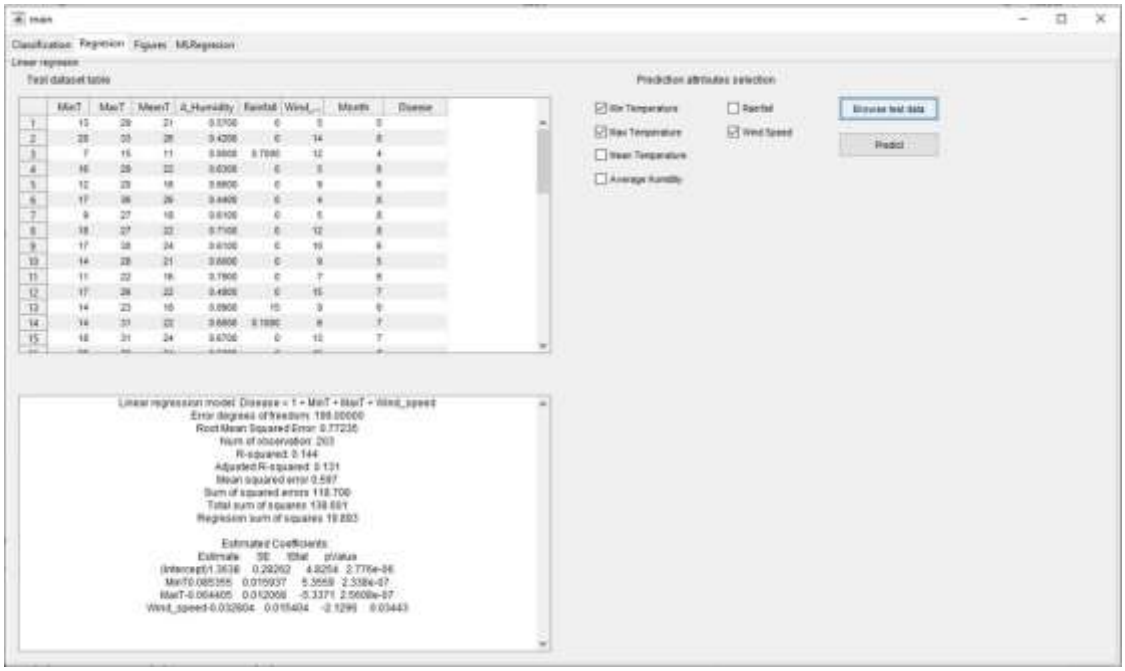
Карактерише га интуитиван интерфејс што омогућава лако учење и коришћење функционалности овог алата. *Weka* подржава више фаза data mining процеса као што је претпроцесирање података, клустеризација, класификација, регресија, визуализација и одабир атрибута. Такође, *Weka* садржи бројне алгоритме машинског учења. На пример, неки од имплементираних алгоритама машинског учења у склопу *Weka* алата су *J48, naive Bayes, Multilayer Percepon, SVM* и други. На овакав начин, Weka укључује цели data mining процес од претпроцесирања до визуализације података. Коришћење *Weka* алата у фази креирања предикционих модела у скопу имплементираног прототипа апликације је значајно због чињенице да су функционалности овог алата интегрисане у финални апликацију намењену крајњим корисницима овог система. Коришћење *MatLab* програмског пакета, као и функционалности имплементираних у склопу *Weka* алата, омогућило је креирање већег броја предикционих модела заснованих, како на математичким методама тако и на data mining методама предикције.

Прототип апликације имплементиран коришћењем *MatLab* програмског пакета се састоји од четири дела. Целокупна интеракција и употреба креираног прототипа апликације је заснована на креираном графичком корисничком инерфејсу. Практично, сваки од четири имплеметирана дела представља једну картицу у оквиру главне *GUI* форме која се покреће са покретањем апликације. Први део имплементираног прототипа апликације представља скуп техника базираних на математичким принципима. Почетни изглед овог дела апликације је дат на слици 50. Изглед овог дела апликације организационо је подељен на два дела. Леви део апликације представља простор у коме се врши приказ тренинг података и који се испуњава након одабира тренинг скупа на основу кога ће се вршити креирање придикционог модела. Испод табеле са подацима и дугмета за одабир тренинг модела, као што се може видети на слици 50, налази се падајући мени из кога је могуће одабрати неки од имплементираних математичких или класификационих метода. Одабиром конкретне математичке или класификационе data mining методе врши се креирање предикционог модела коришћењем већ унетих тренинг података. Десни део апликације почиње одабиром и приказом тест скупа података, након чега следе дугме за одабир тест скупа и дугме за покретање предикције.

Доњи део левог и десног дела дела приказа се завршава простором у коме се врши испис основних вредности грешака, како за податке у тренинг скупу тако и за извршену предикцију. Значење самих грешака, као и њихова улога у евалуацији креираних предикционих модела, биће дато приликом описа поступка евалуације за сваку од група предикционих модела. Приказ тренинг и тест података, као и креирање и евалуација предикционих модела је креиран динамички, што значи да је прилагодљив било ком скупу података, без обзира на број атрибута и број инстанци у скупу податка.

Слика 50: Изглед картице са математичким и класификационим техникама

Други део креираног прототипа апликације обухвата функционалности везане за линеарну регресију. Тачније, овај део прототипа апликације омогућава креирање корелационих модела између вредности независних атрибута и вредности зависног атрибута. Изглед овог дела апликације је дат на слици 51. Оваква имплементација омогућава одабир појединачних атрибута на основу којих ће се извршити креирање предикционог модела. Одабир појединачних атрибута је омогућен помоћу *checkbox*-ова. На овакав начин се предикциони модел може креирати на основу везе једног или више независних атрибута са зависним атрибутом.

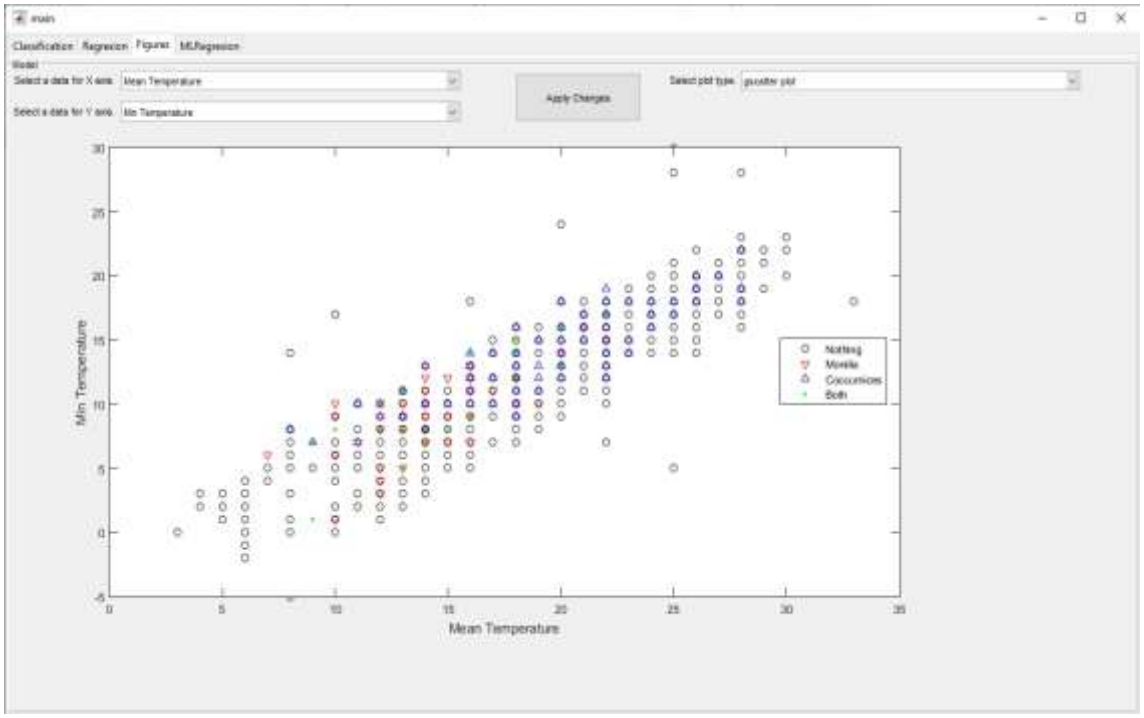


Слика 51: Изглед картице са функционалностима за линеарну регресију

Предикциони модели, креирани на овакав начин, након процеса евалуације могу показати колики утицај поједини независни атрибути имају на сами процес предикције.

Трећи део креираног прототипа апликације обухвата функционалности које омогућавају визуелизацију података. Изглед ове картице након покретања и одабира приказа односа произвољног скупа података је дат на слици 52. Одабиром конкретних атрибута из скупа података омогућава се визуелизација односа одабраних података. Тачније, омогућава се визуелизација целог скупа података на основу односа одабраних атрибута. Визуелизација података се може реализовати кроз више имплементираних метода, одабиром одговарајућег из падајућег менија на десној страни картице.

Различити методи визуелизације омогућавају другачији приказ података, чиме се добија већи број могућности за детекцију *outlier*-а у подацима, као и детекцију међусобних зависности међу подацима. Примера ради, метод *PCA* анализе коришћен за детекцију *outlier*-а је имплементиран у оквиру овог дела прототипа апликације. Такође, када се ради о примени стабала одлучивања, као једних од класификационих техника у домену визуелизације, омогућава се визуелизација креираног стабла одлучивања које практично представља предикциони модел.

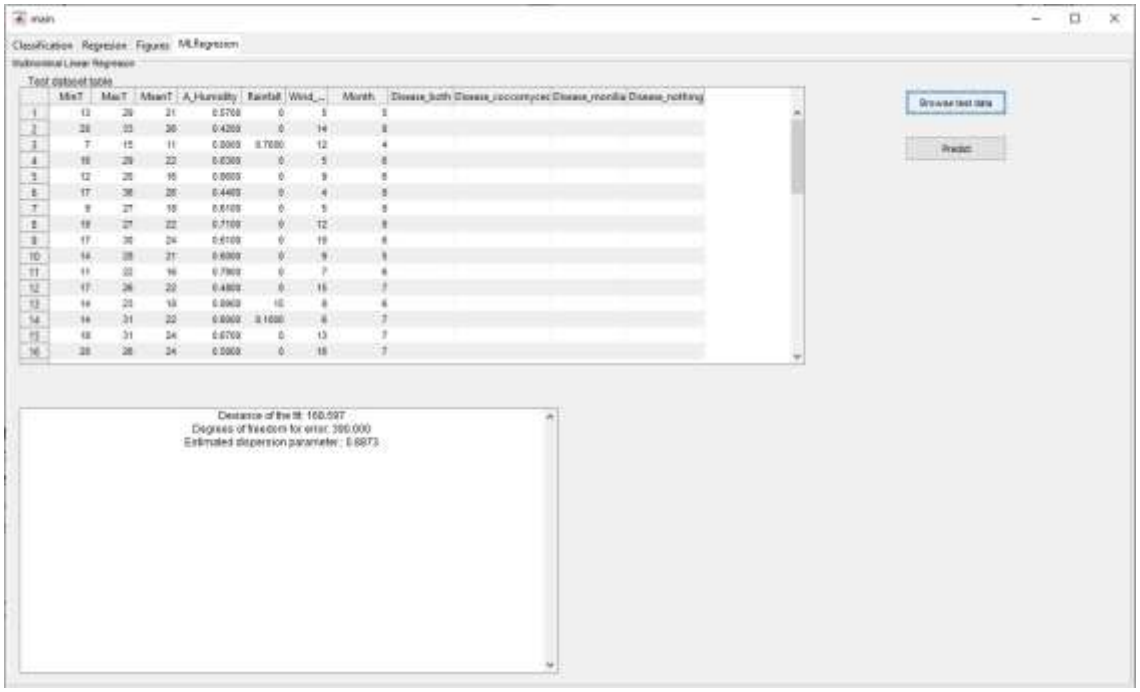


Слика 52: Изглед картице са функционалностима за визуелизацију података

Визуелизацијом креираног стабла одлучивања зависности између података као и параметри гранања постају видљиви, па самим тим постају и лако проверљиви, поређењем са познатим зависностима између независних и зависних података.

Како класни атрибут има више унапред дефинисаних номиналих вредности, извршена је имплементација *multinominal*-не ленеарне регресије. Имплементиране функционалности су груписане у оквиру четвртог дела прототипа апликације који је приказан на слици 53. Централни део овог дела апликације заузима табела са тест подацима за које је потребно одредити вредности класног атрибута. Такође, као и у имплементацијама претходних класификационих метода омогућено је коришћење било ког тест фајла.

Покретањем предикције се, као и код претходних метода, нуди могућност коришћења већ учитаног тренинг скупа података или учитавање и коришћење новог тренинг скупа. Простор испод табеле са тест подацима представља простор у коме се врши приказ статистике изршене предикције у домену тачности предикционих резултата и евидентираног степена грешака.



Слика 53: Изглед картице са функционалностима за multinominal *линерану регресију*

Покретањем предикције за одабрани тест скуп података за сваку од номиналних вредности класног атрибута врши се одређивање могућности да баш та вредност одговара правој вредности класног атрибута. Вредност класног атрибута са највећом вероватноћом се може сматрати предиктивном вредношћу класног атрибута за конкретну инстанцу података.

Улога *Weka* data mining алата у домену креирања и евалуације предикционих модела се огледала у примени класификационих data mining метода имплементираних у склопу овог алата. Као што је раније наведено, функционалности овог алата су интегрисане у финалну верзију софтверског решења намењеног крајњим корисницима. Поред креирања и евалуације предикционих модела *Weka* алат омогућава и детекцију и отклањање *outlier*-а процесом визуелизације података. Такође, у склопу овог алата се могу креирати предикциони модели који се касније могу користити од стране других апликација као већ готови модели. Класификационе data mining технике, имплементиране у склопу овог алата, пружају веома добре перформансе приликом креирања, обуке и коришћења предикционог модела, па самим тим није било потребе за њиховом имплементацијом у склопу прототипа апликације креиране у *MatLab* програмском пакету, већ су коришћене као већ имплементиране у окиру *Weka* алата.

Предности које *Weka* алат доноси у домену визуелизације података са циљем детекције *outlier*-а искоришћене су за потребе детекције *outlier*-а приликом анализе креиране базе података. Прецизност, вредности грешака и предности и недостаци креираних предикционих модела коришћењем класификационих data mining техника описане су у наредним поглављима. Потребно је нагласити да за потребе креирања предикционог модела нису коришћене све доступне класификационе data mining методе. Од укупног броја доступних метода у склопу *Weka* алата одабране су оне методе које према дефинисаном проблему и структури података у тренинг и тест скупу података могу дати задовољавајуће резултате предикције.

Процес евалуације представља основни поступак процене тачности креираног и обученог предикционог модела. Евалуација креираних предикционих модела је вршена, како применом метода унакрсне валидације тако и применом метода поређења стварних вредности класног атрибута у оквиру тест скупа података и вредности класног атрибута добијених предикцијом. Како би се извршило поређење стварних вредности класних атрибута у оквиру тест скупа са вредностима класних атрибута добијених предикцијом приликом креирања тест скупа података, класни атрибути сваке од инстанци података издвојени су у посебан фајл. Практично, тест скуп, као што се може видети на сликама које показују изглед прототипа апликације, не садржи вредности класног атрибута, већ му се након предикције исте додељују за сваку од инстанци података. Поређењем стварних и предикцијом добијених вредности класног атрибута недвосмислено се добија информација о тачности креираног предикционог модела.

8.1 Математички и статистички предикциони модели

Математичке и статистичке методе, имплементиране у склопу креираног прототипа апликације, су коришћене у процесу креирања и обуке конкретних предикционих модела. Након обуке креираних предикционих модела, извршена је евалуација тачности ових модела. Процес евалуације, као што је раније наведено, извршен је на два начина. Први начин представља унакрсну валидацију. Унакрсна валидација предикционих модела спроведена над скупом података са познатим исходом извршена је за 5 и 10 *fold*-ова.

Познато је да унакрсна валидација процену успешности предикционог модела заснива над подацима доступним у оквиру тренинг скупа података, док се разлика између 5 и 10 *fold*-ова огледа у броју група инстанци података које се на почетку унакрсне валидације проглашавају за тест узорак. Резултат унакрсне валидације је у оба случаја видљив кроз низ класификационих грешака и вредности статистичких параметара чијим се тумачењем могу добити прецизније информације о извршеној предикцији.

Поред унакрсне валидације, тачност креираних предикционих модела је добијена коришћењем унапред креираног тест скупа података у коме се налазе инстанце података које нису коришћене у процесу обуке предикционог модела. Вредности параметара унакрсне валидације су дате у табели 15. Математичке и статистичке методе, на основу којих је извршено креирање предикционих модела, дате су у оквиру колона табеле 15.

Наведене методе представљају генерализоване линеарне предикционе моделе. Овако креираним генерализованим линеарним предикционим моделима одговарају наведени параметри унакрсне валидације. Нумеричке вредности параметара унакрсне валидације показују да не постоји велика разлика између валидације извршене са 5 и 10 *fold*-ова. Практично, добијене вредности појединих параметра су веће код унакрсне валидације са 5 *fold*-ова, док су код других параметара веће вредности добијене за унакрсну валидацију са 10 *fold*-ова. Поред овако добијених вредности, унакрсна валидација са 10 *fold*-ова се може сматрати тачнијом због већег броја тест узорка креираних на основу тренинг скупа података.

Први од укупно пет посматраних параметара унакрсне валидације, добијених израчунавањем, је грешка класификације. Грешка класификације представља вредност погрешно класификованих инстанци података и добија се израчунавањем на основу следеће једначине:

$$L = \sum_{j=1}^{n} w_j I\{\hat{y}_j \neq y_j\}$$

Табела 15: Резултати евалуације креираних предикционих модела

		Линеарна дискриминациона анализа	Псеудо квадратна дискриминациона анализа	Диаг линеарна дискриминациона анализа	Диаг квадратна дискриминациона анализа	Псеудо линеарна дискриминациона алнализа
10-*fold* унакрсна валидација	Грешка класификације	0,16200	0,14700	0,24600	0,20800	0,15600
	Binomial deviance	0,22428	0,20902	0,25701	0,23288	0,22379
	Експоненцијални губитак	0,48701	0,46491	0,52319	0,49278	0,48651
	Hinge	0,23119	0,17564	0,28719	0,22501	0,23056
	Квадратни губитак	0,14054	0,12850	0,20100	0,16838	0,13956
5-*fold* унакрсна валидација	Грешка класификације	0,16000	0,14900	0,24900	0,20800	0,16100
	Binomial deviance	0,22530	0,20965	0,25674	0,23145	0,22330
	Експоненцијални губитак	0,48817	0,46552	0,52261	0,49118	0,48591
	Hinge	0,23318	0,17624	0,28519	0,22243	0,22946
	Квадратни губитак	0,14226	0,12990	0,20138	0,16588	0,13880

Грешка класификације представља тежинску функцију погрешно класификованих инстанци, где је $\hat{y}_j$ класна вредност која одговара класи са максималном вредношћу постериор вероватноће. $I\{x\}$ представља индикатор функцију, док w_j представља тежину за инстанцу j. Различити дискриминациони модели, у зависности од имплементираног дискриминационог типа, показују различите вредности класификационе грешке. Највећа вредност класификационе грешке израчуната је за предикциони модел креиран на основу диаг линеарне дискриминационе анализе, док је најмања вредност грешке класификације израчуната за предикциони модел креиран на основу псеудо квадратне дискриминационе анализе.

Binomial deviance, други од параметара унакрсне валидације, представља статистичку доброту поклапања за статистички модел. Примењује се у случајевима када класни атрибут има више могућих вредности. Вредност овог параметра унакрсне валидације добијена је израчунавањем на применом следеће једначине:

$$L = \sum_{j=1}^{n} w_j log\{1 + exp[-2m_j]\},$$

где је w_j тежина за инстанцу *j*, док је m_j скаларни класификациони резултат који модел предвиђа за тачну вредност посматране класе. Ово је генерализација суме квадрата резидуала метода најмањих квадрата примењеног на случајеве када се подешавање модела врши на основу највеће вероватноће. Израчунате вредности овог параметра се разликују за различите предикционе моделе, како за 5-*fold* унаксну валидацију тако и за 10-*fold* унаксну валидацију. Израчунате вредности овог праметара, такође, показују различитост у вредностима за различит број *fold*-ова, па је тако за прва два предикциона модела израчуната вредност мања приликом валидације са 10 *fold*-ова, док је код преостала три предикциона модела израчуната вредност већа за унакрсну валидацију са 10 *fold*-ова. Поред поменутих различитости у израчунатим вредностима овог параметра, најмања вредност овог параметра, како за 5-*fold* тако и за 10-*fold* унакрсну валидацију, израчуната је за предикциони модел креиран на основу псеудо квадратне дискриминационе анализе. Као што се може видети на основу података у табели 15, овај предикциони модел је један од поменута два предикциона модела код којих је вредност овог параметара мања у случају извршења 10-*fold* унакрсне валидације у односу на извршење 5-*fold* унакрсне валидације.

Трећи параметар спроведене унакрсне валидације под називом експоненцијални губитак се израчунава према следећој једначини:

$$L = \sum_{j=1}^{n} w_j \exp(-m_j)$$

Као и у случају *binomial deviance* параметра, w_j представља тежину за инстанцу j, док је m_j скларни класификациони резултат који модел предвиђа за тачну вредност посматране класе. Приликом израчунавања овај параметар строжије поступа са погрешним предвиђањима од осталих параметара, а притом има и већи градијент. Вредности овог параметра за 10-*fold* унакрсну валидацију мање су од вредности за 5-*fold* унакрсну валидацију када се ради о прва два предикциона модела. Када се ради о преостала три предикциона модела ситуација је обрнута, па су вредности овог параметра за случај 10-*fold* унакрсне валидације веће од 5-*fold* унакрсне валидације. Најмања вредност овог параметра је израчуната за предикциони модел креиран на основу псеудо квадратне дискриминационе анализе, док је највећа вредност израчуната за предикциони модел креиран на основу диаг линеарне дискриминационе анализе.

Четврти и предпоследњи параметар унакрсне валидације је познат под називом *Hinge* губитак. Вредност овог параметра се израчунава на основу следеће једначине:

$$L = \sum_{j=1}^{n} w_j max\{0,1 - m_j\}$$

При чему су све променљиве у оквиру дате једначине већ познате. Практично, *hinge* губитак функција обезбеђује релативно уску конвексну горњу границу 0-1 индикатор функцији. Додатно, емпиријска минимизација ризика овог губитка еквивалентна је класичној формулацији *SVM*-а. На овакав начин, коректно класификоване инстанце које се налазе изван маргина вектора подршке се не кажњавају, док се инстанце података унутар маргина или на супротоној страни хиперплејна кажњавају линеарно у односу на њихову удаљеност од коректне границе. Иако је *hinge* губитак као функција у исто време конвексна и непрекидна, ова функција није диференцијабилна. Управо из ових разлога се *hinge* губитак функција не може користити са методима *gradient descent*-а или стохастичким *gradient descent* методима који су по природи диференцијабилни на целом домену.

Вредности овог параметра за спроведену унаксну валидацију, као што је то био случај и са претходним валидационим параметрима, најмање су за предикциони модел базиран на псеудо квадратној дискриминационој анализи, док су највеће за предикциони модел базиран на diag linearnoj дискриминационој анализи.

Последње посматрани параметар на основу кога је вршено поређење креираних предикционих модела јесте квадратни губитак. Израчунавање вредности квадратног губитка се може извршити на основу следеће једначине:

$$L = \sum_{j=1}^{n} w_j (1 - m_j)^2$$

Иако је функција овог губитка најчешће коришћена у регресионим проблемима, веома лако се може применити на класификационе проблеме, што је случај са применом код унакрсне валидације креираних предикционих модела. Ова функција је конвексна и глатка и одговара опсегу од 0 до 1. Предност квадратне функције губитка се огледа у томе што је њена структура погодна за једноставну унакрсну валидацију параметара. Вредности овог параметра за извршену унакрсну валидацију показују да најбоље перформансе као и у претходним случајевима има предикциони модел креиран на основу псеудо квадратне дискриминационе анализе. Најбоље перформансе се огледају у најмањој израчунатој вредности квадратног губитка.

Унакрсна валидација, поред наведених вредности параметара, пружа информације о вредностима средње квадратне грешке, као и вредности стопе погрешне класификације. Посматрано из угла статистике, средња квадратна грешка или средња квадратна девијација је процедура за процену непознатих квантитета која мери средњу вредност квадрата грешака, односно средњу вредност квадрата разлике између стварних вредности класних атрибута и вредности добијених предикцијом. У питању је функција ризика која одговара очекиваној вредности квадрата грешке. Вредности ове функције најчешће су ненегативне, при чему вредности ближе нули показују да се ради о бољем предикционом моделу. Стопа погрешне класификације се понаша као значајна мера у одређивању тога који је од креираних предикционих модела бољи.

Мање вредност овог параметра указују на бољи предикциони модел. Вредност средње квадратне грешке 10-*fold* унакрсне валидације, спроведене над тренинг скупом података са познатим исходом, износи 1,202, док вредност ове грешке за 5-*fold* унакрсну валидацију износи 1,1897. Са друге стране, стопа погрешне класификације износи 0,296 и 0,297 за 10-*fold* і 5-*fold* унакрсну валидацију респективно.

Тачност предикције креираног и обученог предикционог модела један је од најзначајних параметара у процесу одабира одговарајућег модела. Тачност предикционог модела се израчунава стављањем у однос стварних вредности класних атрибута и вредности класних атрибута добијених предикцијом. Један од начина за израчунавање тачности креираног предикционог модела јесте креирање *confusion* матрице. Истовремено, на основу вредности у креираној *confusion* матрици, поред тачности предикционог модела, могу се израчунати вредности већег броја параметара на основу којих се добијају додатне информације о креираном предикционом моделу. Израчунавање *confusion* матрице и статистичких параметара на основу вредности у *confusion* матрици је имплементирано у склопу прототипа апликације. Имплементиране функционалности позиване су за сваки од креираних предикиционих модела у циљу добијања потребних информација за поређење перформанси ових модела. Приликом позива имплементиране функције истој се прослеђују стварне вредности класног атрибута и вредности класног атрибута добијене предикцијом. Уколико се посматрају *confusion* матрице, редови у свакој од матрица представљају стварне вредности класних атрибута, док колоне у матрици представљају вредности класног атрибута добијене предикциојом. Стављањем у однос ових вредности добијене су вредности мера потребних за даље поређење. Основне мере које се добијају на основу вредности у *confusion* матрици су наведене у наставку, с обзиром да су детаљно описане у претходним поглављима.

- TP (енг. *True Positives*). У ову групу се убрајају све инстанце података за које је предикцијом одређено да припадају датој класи, што се поклапа са стварним вредностима класног атрибута.

- TN (енг. *True Negatives*). У ову групу се убрајају све инстанце података за које је одређено да не припадају датој класи, а притом је на основу стварних вредности класног атрубута утврђено да оне заиста не припадају датој класи.
- FP (енг. *False Positives*). У ову групу се убрајају све инстанце података за које је предикцијом одређено да одговарају конкретној класи, док на основу стварних вредности не одговарају тој класи.
- FN (енг. *False Negatives*). У ову групу се убрајају све инстанце података за које је предикцијом одређено да не одговарају конкретној класи, док на основу стварних вредности ипак одговарају тој класи.
- Тачност (енг. *Accuracy*). Описује колико често је предикциони модел правилно класификовао инстанце података.
- Одзив (енг. *Recall*). Другачије се назива и *True Positive Rate.* Уколико дата инстанца заиста припада конкретној класи, овај параметар показује колико често је предикцијом заиста одређено да припада датој класи.
- Специфичност (енг. *Specificity*). Другачије се назива и *True Negative Rate*. Уколико дата инстанца заиста не припада конкретној класи, овај параметар показује колико често је предикцијом и одређено да не припада датој класи.
- Прецизност (енг. *Precision*). Уколико је предикцијом одређено да дата инстанца одговара конкретној класи, колико често је таква предикција тачна.
- F-мера (енг. *F-Score*). Представља просечан однос одзива и прецизности параметра. У статистичкој анализи F-мера представља меру тачности спроведеног теста.

Вредности параметара TP, TN, FP и FN готово су видљиве из саме *confusion* матрице. Притом су вредности ових параметара потребне како би се израчунале вредности осталих наведених параметара. Свака од поменутих вредности се може израчунати, како за појединачну класну вредност тако и за цео скуп података. Примера ради, тачност предикционог модела се са једне стране може израчунати за сваку од класних вредности понаособ, док се са друге стране може израчунати конкретно за цео предикциони модел.

Креиране *confusion* матрице, потребне за даље израчунавање поменутих параметара за сваки од креираних предикционих модела. приказане су у наставку.

Линеарна дис. анализа	*nothing*	*monilia*	*coccomyces*	*both*
nothing	163	4	1	0
monilia	11	1	1	0
coccomyces	22	0	14	0
both	6	0	1	0

Псеудо квадратна дис. анализа	*nothing*	*monilia*	*coccomyces*	*both*
nothing	163	4	1	0
monilia	10	3	0	0
coccomyces	19	0	15	2
both	4	2	0	1

Диаг линеарна дис. анализа	*nothing*	*monilia*	*coccomyces*	*both*
nothing	145	23	0	0
monilia	5	8	0	0
coccomyces	22	3	11	0
both	3	4	0	0

Диаг квадратна дис. анализа	*nothing*	*monilia*	*coccomyces*	*both*
nothing	154	20	2	1
monilia	4	9	0	0
coccomyces	17	0	13	6
both	2	3	0	2

Псеудо линеарна дис. анализа	*nothing*	*monilia*	*coccomyces*	*both*
nothing	163	4	1	0
monilia	11	1	1	0
coccomyces	22	0	14	0
both	6	0	1	0

Поређењем креираних *confusion* матрица се уочава да су ове матрице за први и последњи предикциони модел истоветне, на основу чега се могу очекивати исте вредности параметра ова два модела. Оваква једнакост није евидентирана код изршене унакрсне валидације, што се може видети прегледом вредности у табели 15. Вредности поменутих параметара, добијене израчунавањем на основу вредности у датим *confusion* матрицама за сваки од креираних предикционих модела, дате су у оквиру табеле 16.

Као што се може видети, вредности за TP, TN, FP и FN нису приказане у табели 16, јер се ове вредности, као што је раније наведено, користе као међурезултати за израчунавање вредности параметара приказаних у табели 16.

Табела 16: Поређење предикционих модела на основу вредности израчунатих из confusion *matrice*

Параметри	Класне вредности	Линеарна дискриминациона анализа	Псеудо квадратна дискриминациона анализа	Диаг линеарна дискриминациона анализа	Диаг квадратна дискриминациона анализа	Псеудо линеарна дискриминациона алнализа
Тачност	nothing	0,8036	0,8304	0,7634	0,7946	0,8036
	monilia	0,9286	0,9286	0,8438	0,8795	0,9286
	coccomyces	0,8884	0,9018	0,8884	0,8884	0,8884
	both	0,9688	0,9643	0,9688	0,9464	0,9688
Одзив	nothing	0,9702	0,9702	0,8631	0,8631	0,9702
	monilia	0,0769	0,2308	0,6154	0,6923	0,0769
	coccomyces	0,3889	0,4167	0,3056	0,3611	0,3889
	both	0	0,1429	0	0,2857	0
Специфи;ност	nothing	0,3036	0,4107	0,4643	0,5893	0,3036
	monilia	0,9810	0,9716	0,8578	0,8910	0,9810
	coccomyces	0,9840	0,9947	1	0,9894	0,9840
	both	1	0,9908	1	0,9677	1
Прецизност	nothing	0,8069	0,8316	0,8286	0,8631	0,8069
	monilia	0,2000	0,3333	0,2105	0,2813	0,2000
	coccomyces	0,8235	0,9375	1	0,8667	0,8235
	both	NaN	0,3333	NaN	0,2222	NaN
F-мера	nothing	0,8811	0,8956	0,8455	0,8631	0,8811
	monilia	0,1111	0,2727	0,3137	0,4000	0,1111
	coccomyces	0,5283	0,5769	0,4681	0,5098	0,5283
	both	0	0,2000	0	0,2500	0
Тачност модела		0,7946	0,8125	0,7321	0,7545	0,7946

Порђењем вредности свих израчунатих грешака и губитака у оквиру табеле 15, као и вредности параметара у оквиру табеле 16. добија се довољан број информација о перформансама сваког од креираних предикционих модела.

На основу свих вредности наведених у табели 15, предикциони модел креиран коришћењем класификатора заснованог на принципима псеудо квадратне дискриминационе анализе, издаја се као најбоље решење у овој групи предикционих модела. Уколико се посматрају параметри у табели 16, а посебно параметар тачности предикционог модела, псеудо квадратна дискриминациона анализа се и у овом случају издваја као метода чијим се коришћењем може креирати предикциони модел са најбољим перформансама. Имајући у виду целокупан спроведени процес евалуације креираних предикционих модела, може се закључити да је псеудо квадратна дискриминациона анализа најпогоднији метод за креирање предикционог модела из групе генерализованих линеарних модела.

8.2 Класификациони предикциони модели

Креирање предикционих модела базираних на скупу података са познатим исходом се може извршити применом различитих класификационих data mining алгоритама. *Weka* data mining алат, као што је на почетку овог поглавља наведено, садржи, између осталог, велики број класификационих data mining техника које се могу применити у поступку креирања предикционих модела. Доступност већег броја класификационих алата само је један од разлога интеграције *Weka* библиотеке, како у креираном прототипу апликације тако и у финалној апликацији намењеној крајњим корисницима. Класификационе data mining технике, погодне за креирање и евалуацију предикционих модела у оквиру *Weka* алата, су подељене у већи број категорија, при чему свака од категорија класификационих техника групише сличне технике. Како би се одредило које од техника у оквиру сваке од група задовољавају критеријуме потребне за успешну предикцију, извршена је евалуација већег броја класификационих техника у оквиру сваке од група. Евалуација класификационих техника је вршена на основу унапред дефинисаног скупа грешака и статистичких параметара. На овакав начин су из сваке од група издвојени најадекватнији представници. Овако одабране класификационе технике чине финални скуп техника и даље се користе у поступку креирања предикционих модела. Сваки од креираних предикционих модела, са циљем одређивања тачности, евалуиран је на већ утврђен и дефинисан начин.

Као што је то био случај са примењеним математичким и статистичким техникама креирања предикционих модела и у случају класификационих техника је најпре вршена 10-*fold* и 5-*fold* унакрсна валидација коришћењем целокупног тренинг скупа података. Резултати извршене унакрсне валидације за одабрани број *fold-ова* су приказани у оквиру табеле 17.

Табела 17: Резултати евалуације креираних класификационих предикционих модела

		Bayes Net	Multilayer Perceptron	KStar	Classification Via Regression	PART	J48	Random Forest
10-*fold* унакрсна валидација	Kappa statistic	0,6192	0,6079	0,6549	0,7728	0,8013	0,7980	0,8063
	Mean absolute error	0,0825	0,0877	0,0660	0,0602	0,0414	0,0433	0,0516
	Root mean squared error	0,2287	0,2250	0,2068	0,1664	0,1750	0,1808	0,1561
	Relative absolute error	0,4616	0,4906	0,3694	0,3367	0,2314	0,2423	0,2878
	Root relative squared error	0,7669	0,7545	0,6935	0,5582	0,5868	0,6064	0,5236
5-*fold* унакрсна валидација	Kappa statistic	0,6121	0,6382	0,6373	0,7828	0,7837	0,7683	0,8068
	Mean absolute error	0,0840	0,0843	0,0671	0,0602	0,0446	0,0464	0,0523
	Root mean squared error	0,2300	0,2151	0,2086	0,1654	0,1816	0,1852	0,1565
	Relative absolute error	0,4699	0,4716	0,3753	0,3366	0,2495	0,2597	0,2926
	Root relative squared error	0,7714	0,7212	0,6996	0,5547	0,6089	0,6213	0,5248

Практично, у оквиру табеле 17 су приказани евалуациони резултати најбољих класификатора за сваку од класификационих група, што уједно представља и поређење одабраних класификатора. Називи одабраних класификатора су наведени по колонама табеле 17, док су евалуациони параметри, на основу којих се врши поређење, дати по редовима. Наведени валидациони параметри су добијени у форми извештаја, након извршене унакрсне валидације, па су ради лакшег разумевања наведени са оригиналним називима на енглеском језику. Сваки од наведених валидационих параметара у наставку текста биће детаљније објашњен.

Први у низу посматраних параметара јесте *Kappa* статистика. Овај параметар представља хеуристички начин генерализације алгоритама који описују вишекласне проблеме. *Kappa* статистика се користи за мерење поклапања између предикцијом добијених и посматраних категоризација скупа података, док се корекција врши за поклапања у случајевима који се случајно догоде. Коректно или некоректно класификоване инстанце дефинишу случај у коме су инстанце коришћење као тест подаци. Посматрано на овакав начин, *Kappa* статистика представља споразум нормализован за случајни погодак и може се израчунати применом следеће једначине [126].

$$Kappa = \frac{P(A) - P(E)}{1 - P(E)}$$

У оквиру наведене једначине *P(A)* представља процентуално поклапање (поклапање између класификованих и тачних резултата), док *P(E)* представља случајни погодак. Вредност *Kappa=1* указује на савршено поклапање, док вредност *Kappa=0* указује на случајни погодак. Поређењем добијених вредности *Kappa* статистике за сваки од одабраних класификатора се може видети да приближно исте и уједно и највеће вредности имају последње наведена три модела, како за 10-*fold* тако и за 5-*fold* унакрсну валидацију.

Други посматрани параметар је просечна апсолутна грешка (енг. *Mean absolute error - MAE*). Ова грешка представља само просечну величину појединачних грешака у предикционом скупу без разматрања њиховог правца [127].

Тачније, ова грешка представља просечну вредност тест узорка апсолутних разлика између вредности добијене предикцијом и стварне вредности, где су све индивидуалне разлике једнаке тежине. Просечна апсолутна грешка се израчунава применом следеће једначине:

$$MAE = \frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{n}$$

Где p и a представљају предикцијом добијену вредност и стварну вредност класног атрибута респективно, док n представља број инстанци у скупу података. Поређењем вредности добијених приликом извршења 10-*fold* и 5-*fold* унакрсне валидације може се видети да последња три модела имају најмању вредност ове грешке. Притом, за предикциони модел, базираним на *PART* класификационом алгоритму, евидентна је најмања вредност грешке.

Трећи посматрани параметар је корена грешка средњег квадрата (енг. *Root mean squared error - RMSE*) и представља квадратни корен просечног квадратног губитка. Вредност овог параметра класификације се може израчунати применом следеће једначине:

$$RMSE = \sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}}$$

Као и у претходном случају, p и a представљају предикцијом добијену вредност и стварну вредност респективно, док n представља број инстаници података. *RMSE* је стандардна девијација резидуала. Резидуали представљају меру растојања датог податка од регресионе линије. *RMSE* је мера која показује колико су у простору распрострањени резидуали. Односно, овај параметар показује како су подаци концентрисани око линије која представља најбоље поклапање [127].

Поређењем израчунатих вредности ове грешке, како за случај 5-*fold* унакрсне валидације тако и за случај 10-*fold* унакрсне валидације, најбоље резултате показује модел креиран применом *Random Forest* алгоритма.

Реалтивна апсолутна грешка (енг. *Relative absolute error - RAE*) представља нормализацију укупне апсолутне грешке. Нормализација се врши на основу грешке једноставног предиктора који врши предикцију просечних вредности. Вредности ове грешке се могу израчунати применом следеће једначне:

$$RAE = \frac{|p_1 - a_1| + \cdots + |p_n - a_n|}{|a_1 - \bar{a}| + \cdots + |a_n - \bar{a}|}$$

Ознаке у оквиру дате једначине представљају истоветне вредности, као што је то био случај са ознакама у претходно поменутим параметрима. Додатно $\bar{a} = \frac{1}{n}\sum_i a_i$. Ова грешка је релативна према једноставном предиктору који представља просек стварних вредности. Тачније, релативна апсолутна грешка узима вредност укупне апсолутне грешке датог предиктора и нормализује је дељењем са вредношћу укупне апсолутне грешке једноставног предиктора. Када је у питању поменута грешка, поређењем добијених вредности спроведених унакрсних валидација за сваки од наведених класификатора, може се уочити да последња три класификатора имају најмању и приближно исту вредност ове грешке.

Корена релативна квадратна грешка (енг. *Root relative squared error - RRSE*) је релативна у односу на оно што би било да се користи једноставан предиктор. Вредност ове грешке се може израчунати на основу следеће једначине:

$$RRSE = \sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{(a_1 - \bar{a})^2 + \cdots + (a_n - \bar{a})^2}}$$

Једноставни предиктор представља само просечну вредност стварних вредности. На овакав начин, релативна квадратна грешка узима вредност укупне квадратне грешке и врши њену нормализацију дељењем са вредношћу укупне квадратне грешке једноставног предиктора. Узимајући квадратни корен релативне квадратне грешке врши се редукција ове грешке на исте димензије као што је квантитативно предвиђено. Поређењем вредности ове грешке, датих у оквиру табеле 17, евидентно је да је најмања вредност грешке добијена у случају примене *Random Forest* класификатора, како за 5-*fold* тако и за 10-*fold* унакрсну валидацију.

Евалуација креираних предикционих модела, поред израчунавања вредности поменутих параметара грешака, као што је то био случај са претходном групом предикционих модела, захтева и израчунавање тачности самих модела. За сваки од креираних предикционих модела заснованих на класификационим техникама извршено је тестирање поклапања класних вредности добијених предикцијом и стварних вредности за дати тест скуп података. Као што је то био случај и са претходном групом предикционих модела, и у овом случају су креиране *confusion* матрице за сваки од предикционих модела. Вредности у оквиру креираних *confusion* матрица коришћене су у процесу израчунавања статистичких параметара неопходних за успешно поређење предикционих модела. Редови у свакој од *confusion* матрица представљају стварне вредности класних атрибута, док колоне у матрици представљају вредности класног атрибута добијене предикцијом. Назив класификационе технике примењене на креирање предикционог модела коме одговара конкретна *confusion* матрица дат је у првом пољу *confusion* матрице. Вредности *TP*, *TN*, *FP* и *FN* параметара се добијају директним очитавањем (*TP* вредности су вредности на главној дијагонали) или израчунавањем за сваки од класних атрибута.

BayesNet	*nothing*	*monilia*	*coccomyces*	*both*
nothing	150	15	3	0
monilia	1	10	2	0
coccomyces	6	4	24	2
both	0	4	2	1

Multilayer Perceptron	*nothing*	*monilia*	*coccomyces*	*both*
nothing	161	5	2	0
monilia	6	5	2	0
coccomyces	10	0	26	0
both	3	0	4	0

KStar	*nothing*	*monilia*	*coccomyces*	*both*
nothing	164	2	1	1
monilia	6	6	1	0
coccomyces	7	1	28	0
both	1	4	2	0

ClassificationViaRegression	*nothing*	*monilia*	*coccomyces*	*both*
nothing	165	2	1	0
monilia	3	7	3	0
coccomyces	2	0	34	0
both	0	2	5	0

PART	*nothing*	*monilia*	*coccomyces*	*both*
nothing	163	4	1	0
monilia	3	8	2	0
coccomyces	4	1	31	0
both	1	1	2	3

J48	*nothing*	*monilia*	*coccomyces*	*both*
nothing	163	2	3	0
monilia	3	5	3	2
coccomyces	1	0	33	2
both	0	3	3	1

Random Forest	*nothing*	*monilia*	*coccomyces*	*both*
nothing	164	2	2	0
monilia	3	5	3	1
coccomyces	0	0	34	2
both	0	3	3	2

Такође, и у овом случају вредности *TP*, *TN*, *FP* и *FN* параметара, као што је раније наведено, користе се као међурезултати за израчунавање тражених вредности параметара. На основу ових вредности се врши израчунавање вредности параметара, а на основу којих се може вршити процена креираних предикционих модела. У случају класификационих предикционих модела, као што је то био случај и са претходном групом модела, вршено је израчунавање следећих параметара: тачност, одзив, специфичност, прецизност и *F*-мера. Вредности сваког од поменутих параметара, на основу *confusion* матрице, су израчунате за сваку од класних вредности засебно. Последњи у низу параметара потребних за поређење већег броја предикционих модела и процену успешности креираних предикционих модела јесте тачност целокупног модела.

Преглед израчунатих вредности параметара за сваки од класних атрибута, као и укупна тачност сваког од креираних предикционих модела, добијене израчунавањем на основу вредности у креираним *confusion* матрицама, дате су у оквиру табеле 18.

Називи класификационих техника које су коришћене за креирање појединачних предикционих модела наведене су у колонама табеле 18, док су посматрани параметри наведени у редовима дате табеле.

Табела 18: Поређење класификационих предикционих модела на основу вредности израчунатих из confusion *матрице*

Параметри	Klasne vrednosti	Bayes Net	Multilayer Perceptron	KStar	Classification Via Regression	PART	J48	Random Forest
Тачност	nothing	0,8884	0,8839	0,9196	0,9643	0,9420	0,9687	0,9643
	monilia	0,8839	0,9420	0,9750	0,9554	0,9509	0,9464	0,9554
	coccomyces	0,9152	0,9196	0,9464	0,9509	0,9554	0,9554	0,9554
	both	0,9643	0,9688	0,9643	0,9688	0,9821	0,9598	0,9643
Одзив	nothing	0,8930	0,9580	0,9760	0,9820	0,9700	0,9760	0,9760
	monilia	0,7690	0,3850	0,4620	0,5380	0,6150	0,4170	0,6150
	coccomyces	0,6670	0,7220	0,7780	0,9440	0,8610	0,9440	0,9440
	both	0,8260	0	0	0	0,4290	0,2500	0
Специфи;ност	nothing	0,8750	0,6607	0,7500	0,9107	0,8572	0,9464	0,9286
	monilia	0,8910	0,9763	0,9668	0,9810	0,9716	0,9764	0,9764
	coccomyces	0,9628	0,9574	0,9787	0,9521	0,9734	0,9574	0,9574
	both	0,9908	1	0,9954	1	1	0,9861	0,9954
Прецизност	nothing	0,9550	0,8940	0,9210	0,9710	0,9530	0,9820	0,9760
	monilia	0,3030	0,5000	0,4620	0,6360	0,5710	0,5000	0,6150
	coccomyces	0,7740	0,7650	0,8750	0,7910	0,8610	0,8100	0,8100
	both	0,3330	NaN	0	NaN	1	0,4000	0
F-мера	nothing	0,9230	0,9250	0,9480	0,9760	0,9620	0,9790	0,9760
	monilia	0,4350	0,4350	0,4620	0,5830	0,5930	0,4550	0,6150
	coccomyces	0,7160	0,7430	0,8240	0,8610	0,8610	0,8720	0,8720
	both	0,2000	NaN	0	NaN	0,6000	0,3080	0
Тачност модела		0,8258	0,8571	0,8839	0,9196	0,9151	0,9151	0,9196

Сваки од наведених параметара пружа довољно података о томе колико су креирани предикциони модели успешни када се ради о предикцији нових вредности класних атрибута за посматрани скуп података. Поређењем добијених вредности посматраних параметара се уочава да су вредности ових параметара приближно исте када се ради о последње наведена четири модела у табели 18. Остварена тачност предикције ових модела сврстава их у групу адекватних за примену у реалним условима. Који од креираних предикционих модела ће у практичној примени дати најбоље резултате зависи од конкретног скупа података над којим се примењују.

9. Креирани *GreenLife* софтверски систем

Предикција времена хемијских третмана, заснована на скупу података са познатим исходом, као и на примени адекватно креираних и обучених предикционих модела, у многоме може унапредити процес хемијске заштите биљних култура. Креирани прототип апликације, као што је описано у претходном поглављу, показује да је на основу скупа података са познатим исходом могуће успешно извршити предикцију остварености услова за појаву двеју посматраних биљних болести. У исто време, имплементирани предикциони алгоритми у оквиру прототипа апликације показују да је могуће успешно извршити предикцију применом већег броја предикционих алгоритама.

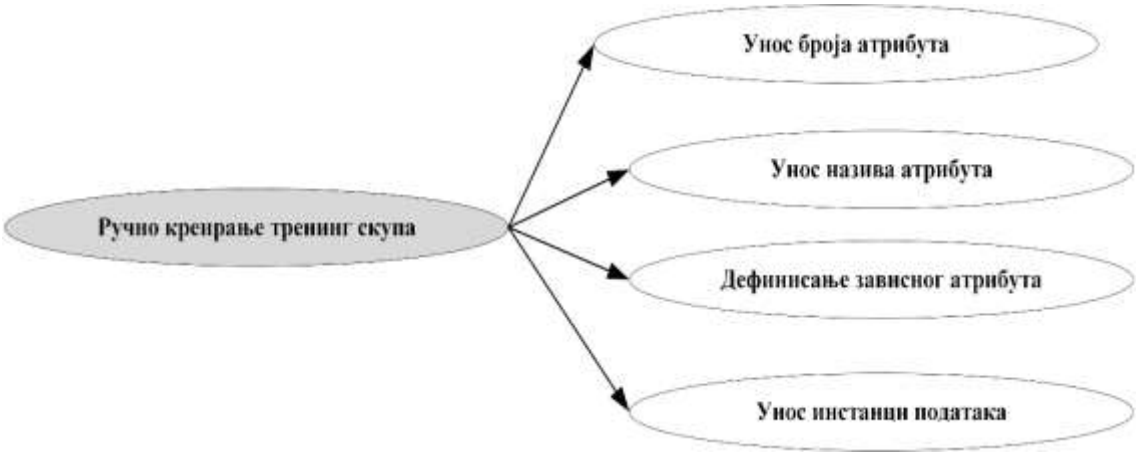
9.1 Случајеви коришћења креираног софтверског система

Тестирањем у склопу прототипа апликације добијени су резултати који показују да су остварени услови за имплементацију посматраних предикционих алгоритама у оквиру софтверског решења намењеног за потребе коришћења од стране крајњих корисника. Развој софтверског решења под називом *GreenLife*, у оквиру кога је имплементиран процес прикупљања података са метеоролошких станица, креирања тренинг и тест скупова података, обука предикционих модела и њихово коришћење у реалним ситуацијама, захтевао је дефинисање корисника оваквог решења, као и случајеве коришћења. Основни *UML* дијаграм који показује кориснике и случајеве коришћења *GreenLife* система приказан је на слици 54.

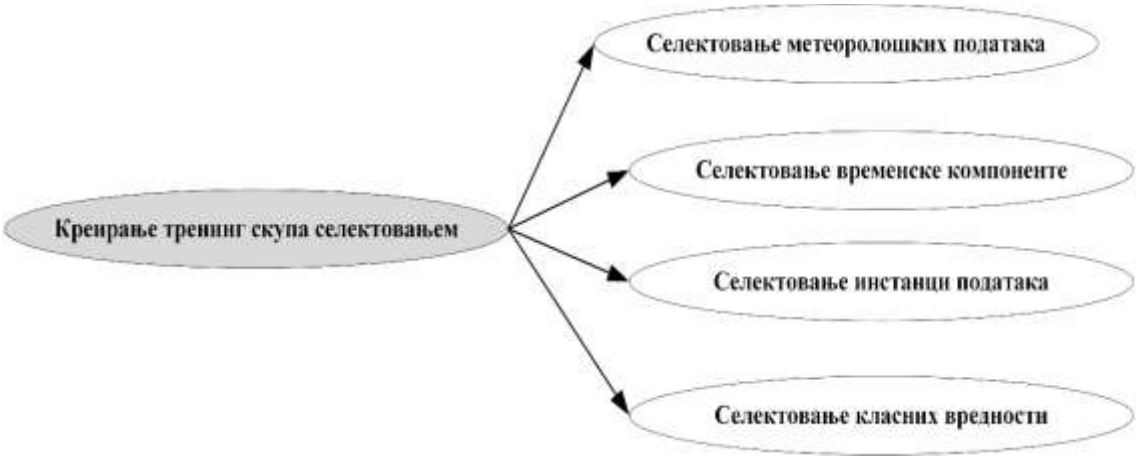
Слика 54: Преглед корисника и случајева коришћења GreenLife система

Већина наведених случајева коришћења обухвата сложеније радње које се даље могу разложити на појединачне случајеве коришћења. Креирање тренинг скупа података са познатим исходом се може обавити ручно у случајевима када очитавање података са метеоролошких станица није аутоматизовано или када се ови подаци за претходни временски период налазе у форми папирних извештаја.

Такође, креирање тренинг скупа података се може обавити и селекцијом у случају постојања креиране базе података са метеоролошким и просторно-временским подацима. Детаљни *UML* дијаграми за случај коришћења *ручно креирање тренинг скупа података* и случај коришћења *креирање тренинг скупа података селекцијом* дати су на слици 55 и слици 56 респективно.



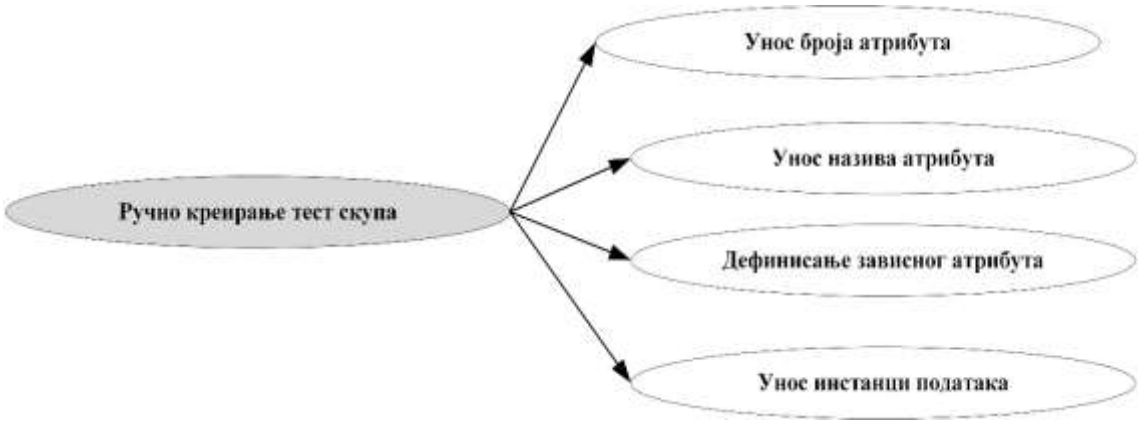
Слика 55: Креирање тренинг скупа података ручним уносом података



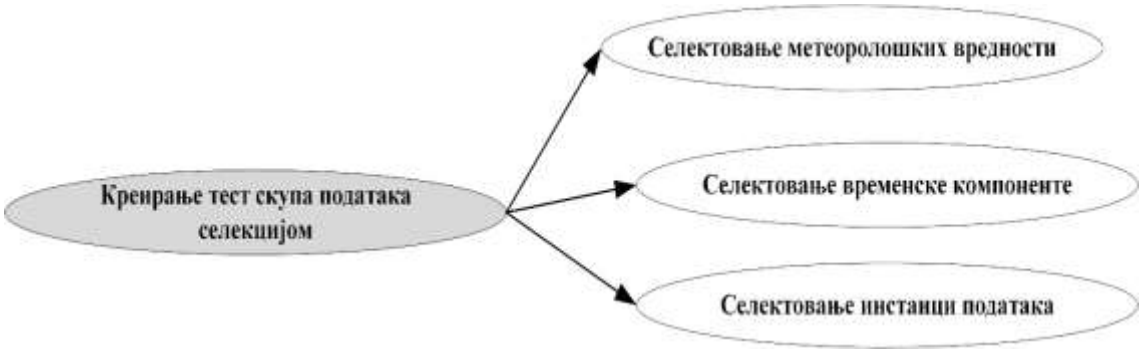
Слика 56: Креирање тренинг скупа података селекцијом података из базе података

Креирање тест скупа података се односи на скуп података за који је потребно предикцијом одредити вредности класног атрибута. Као што је то био случај са креирањем тренинг скупа, и креирање тест скупа података се може обавити на два начина.

Управо из ових разлога су дефинисана два случаја коришћења: *ручно креирање тест скупа података* и *креирање тест скупа података селекцијом*. Детаљни *UML* дијаграми коришћења ова два случаја коришћења су приказани на слици 57 и слици 58 респективно.

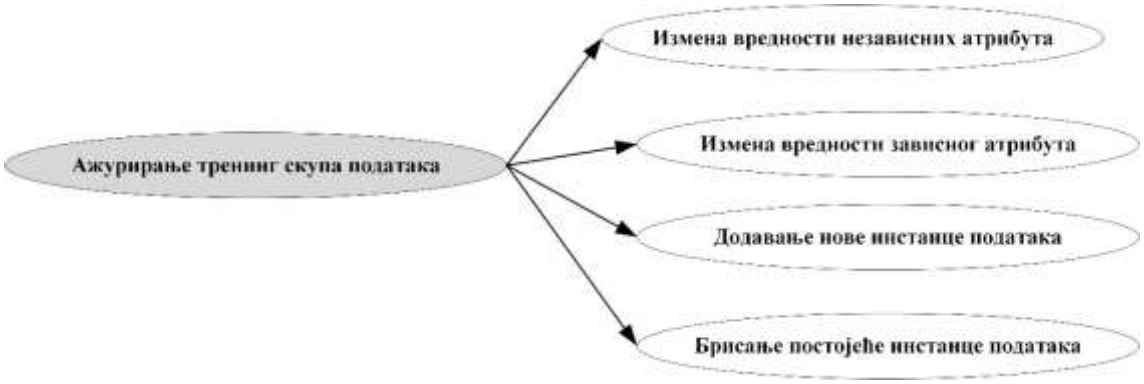


Слика 57: Креирање тест скупа података ручним уносом података

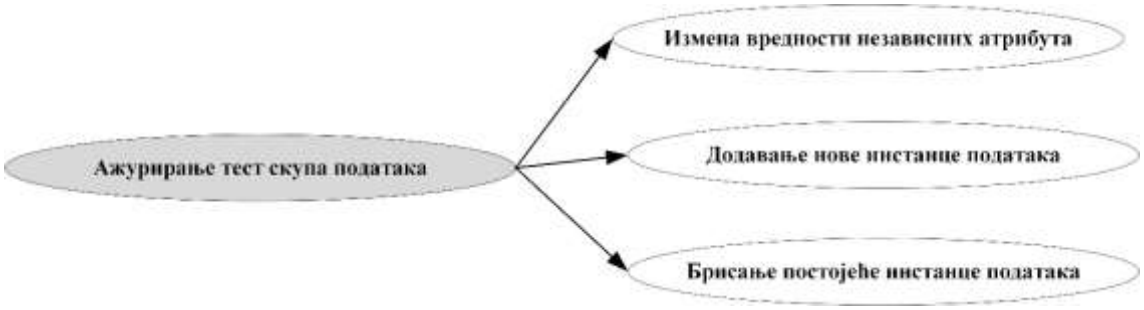


Слика 58: Креирање тест скупа података селекцијом података из базе података

Тренинг и тест скупови података, приликом коришћења у оквиру *GreenLife* система, могу се мењати. Измене тренинг и тест скупа података се односе на тренутне вредности инстанци података у оквиру ових скупова података. Детаљни *UML* дијаграми случајева коришћења, *ажурирање тренинг скупа података* и *ажурирање тест скупа података* дати су на слици 59 и слици 60 респективно.



Слика 59: Ажурирање тренинг скупа података



Слика 60: Ажурирање тест скупа података

Подршка раду са подацима, као и организацији података потребних за рад *GreenLife* софтверског решења, обезбеђена је у виду креирања сопствене базе података. Унос података у оквиру базе података се може обавити ручно од стране корисника или аутоматски, уколико је софтверско решење умрежено са удаљеном метеоролошком станицом или системом метеоролошких станица. Детаљни *UML* дијаграм случаја коришћења *CRUD базе података* дат је на слици 61.

Слика 61: UML дијаграм CRUD базе података

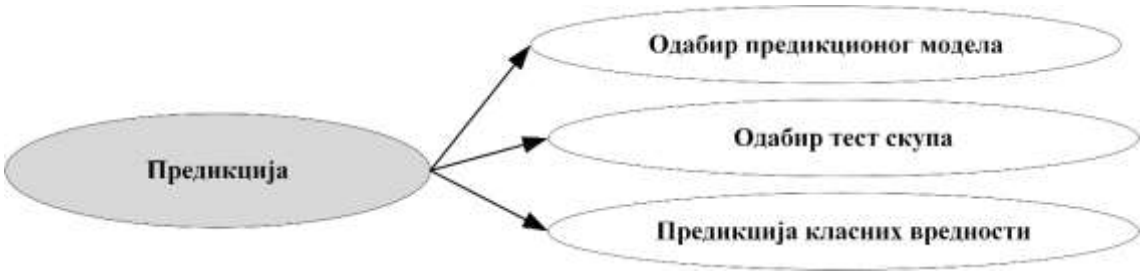
Приказани случај коришћења обухвата креирање базе података, читање из базе података, измене базе података и брисање у оквиру базе податка. Измене базе података се огледају, како у додавању нових инстанци података тако и у изменама постојећих вредности.

Прикупљање података путем бежичног умрежавања са удаљеном метеоролошком станицом или системом метеоролошких станица представља додатне активности корисника система. Детаљни *UML* дијаграм случаја коришћења *комуникација са метеоролошком станицом* дат је на слици 62.

Слика 62: Процес комуникације са метеоролошком станицом и прикупљање података

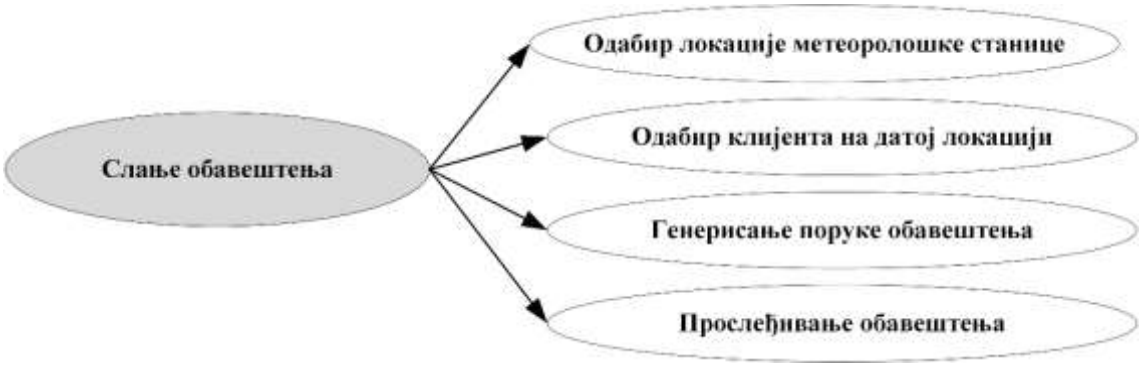
Примена *GreenLife* софтверског решења у сврху предикције времена хемијских третмана се заснива на низу активности које корисник мора обавити како би добио повратну информацију о испуњености услова за појаву посматраних болести. Почетни процес представља креирање тренинг и тест скупа података за чије случајеве коришћења су раније приказани *UML* дијаграми. Након креирања тренинг и тест скупа података се могу обавити случајеви коришћења *обука предикционог модела* и *предикција*. Како ова два случаја коришћења представљају сложне случајеве коришћења детаљни *UML* дијаграми приказани су на слици 63 и слици 64 за случај коришћења *обука предикционог модела* и случај коришћења *предиција* респективно.

Слика 63: Обука предикционог модела на основу скупа података са познатим исходом



Слика 64: Предиција вредности класног атрибута за инстанце података у тест скупу података

Имплементирано *GreenLife* софтверско решење кориснику нуди функционалности потребне за реализацију обавештавања циљне групе пољопривредних произвођача (клијената) о остварености услова за појаву болести, као и правилном времену за хемијску заштиту. Детаљни *UML* дијаграм за случај коришћења *слање обавештења* дат је на слици 65.

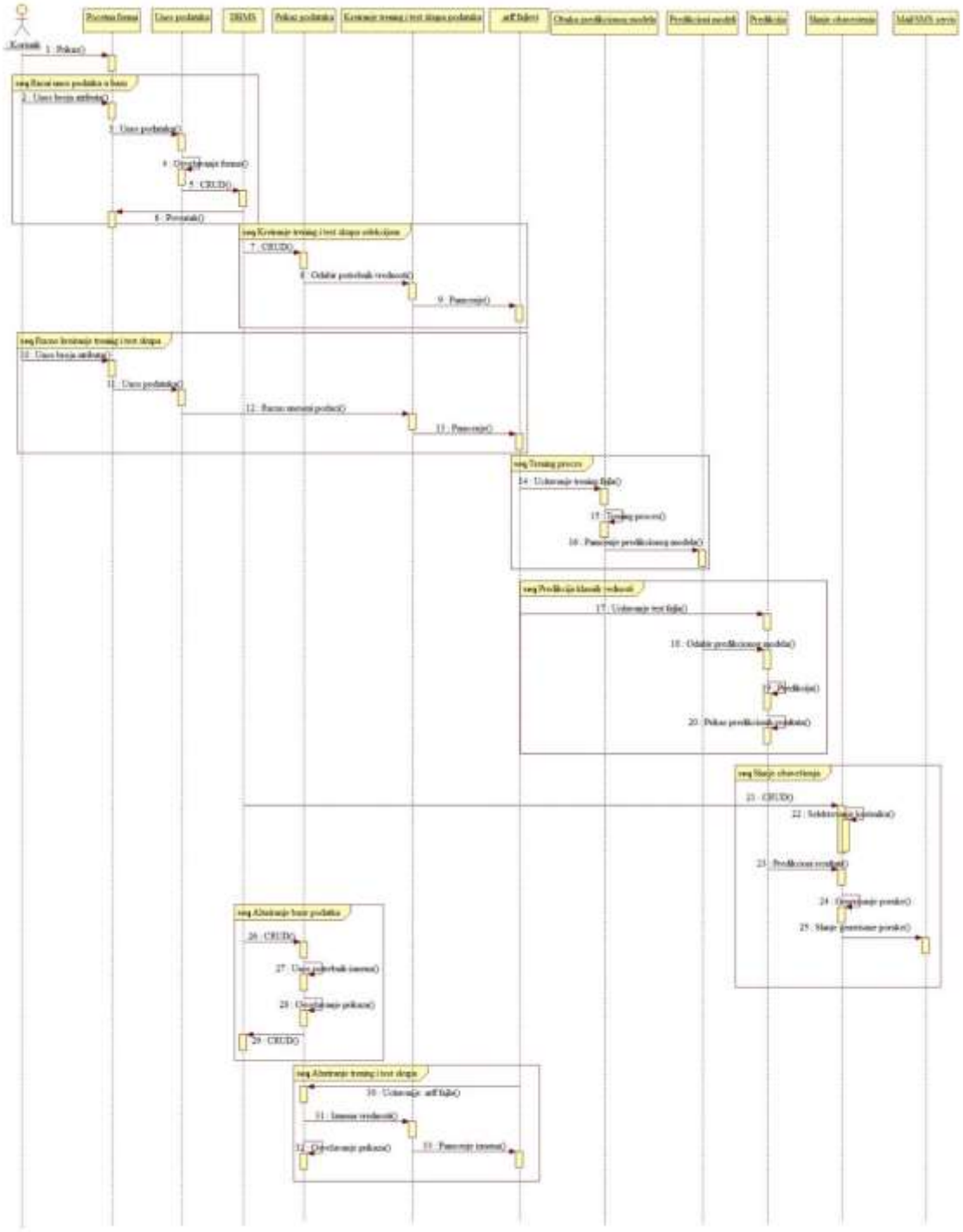


Слика 65: Слање обавештења о добијеним резултатима предикције

Актер сваког од наведених случајева коришћења је корисник *GreenLife* софтверског система. Креирано софтверско решење је намењено, како индивидуалним корисницима тако и организацијама и институцијама.

Индивидуалним корисником, крајњим корисником имплементираног софтверског решења, може се сматрати пољопривредни произвођач, научни радник или особа која се индивидуално бави пружањем услуга дијагнозе и прогнозе појаве болести, као и прогнозом временског распореда хемијских третмана. Допринос имплементираног софтверског решења се из угла индивидуалног корисника огледа у предикцији времена хемијских третмана на основу унетих података за конкретно посматрано подручје. Уколико се индивидуалним корисником сматра пољопривредни произвођач, на основу унетих вредности параметара за свој локалитет или производну површину, пољопривредни произвођач може добити информације о томе када хемијска заштита гајених биљака може имати највећи ефекат. Индивидуални корисник, у случају посматрања једног локалитета, користи најмање могуће ресурсе софтверског решења. Са друге стране, корисници имплементираног софтверског решења могу бити и организације попут хемијских кућа, института или пољопривредне стручне и саветодавне службе. Уколико се ради о коришћењу софтверског решења од стране организација, исте могу на основу метеоролошких и просторно-временских параметара прикупљених са већег броја локација вршити предикцију времена хемијских третмана и обавештавати своје клијенте о правом моменту заштите. Организације могу, на основу добијених података, вршити потребна праћења и предикцију остварености услова за појаву биљних болести коришћењем праћења метеоролошких параметара системом умрежених метеоролошких станица. На основу добијених резултата предикције, примењених за конкретно посматрано подручје употребом имплементираних функционалности, може се издвојити група клијената који припадају датом подручју. На овакав начин се клијенти који имају производне површине на локалитету за који је предикција показала оствареност услова за појаву болести обавештавају коришћењем функционалности имплементираних у оквиру софтверског решења о томе када је правилан моменат за спровођење хемијских третмана.

Сваки од наведених случајева коришћења се карактерише низом операција које корисник система извршава у циљу реализације конкетног задатка. Распоред извршења операција дефинисаних од стране корисника приказан је на слици 66.



Слика 66: Секвенцни дијаграм GreenLife софтверског решења

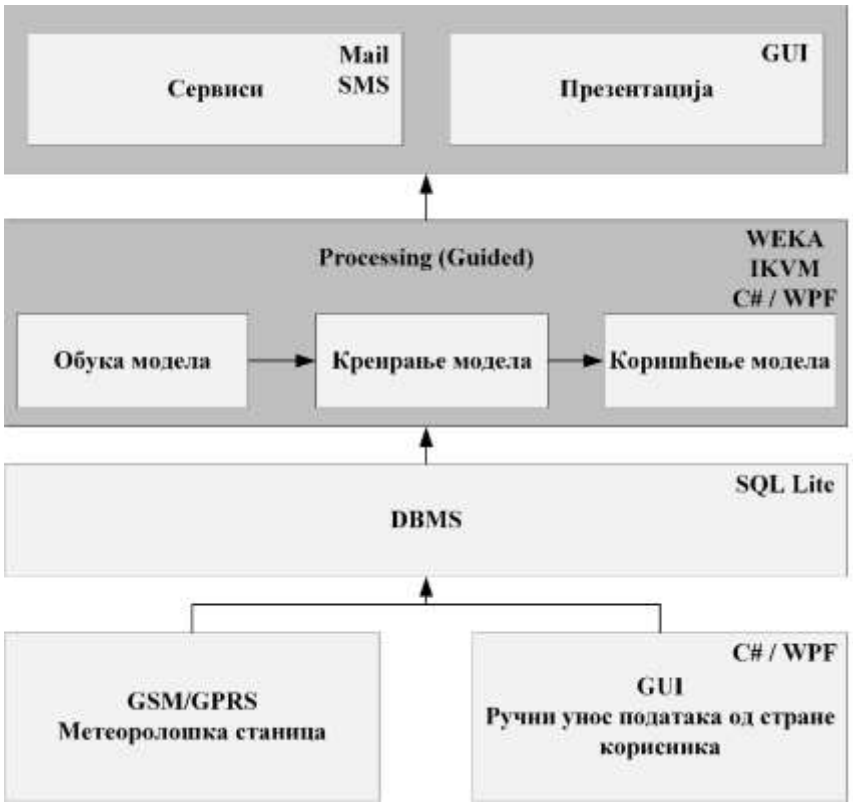
Уколико се посматра креирани секвенцни дијаграм, уочава се да иницарање акције коришћења *GreenLife* софтверског система чини корисник. Такође, иницијално коришћење система започиње се уносом података у базу података.

Након уноса података у базу података се отвара могућност креирања тренинг и тест фајлова. Кориснику је остављена могућност креирања тренинг и тест скупова података без претходног уноса предметних података у базу података. На овакав начин су реализовни случајеви коришћења: *ручно уношење тренинг података* и *ручно уношење тест података*. Корисник може креирати привремени тренинг скуп података на основу којег ће извршити обуку и креирање модела, или са друге стране, корисник система може креирати привремени тест скуп података у коме ће се наћи привремене инстанце метеоролошких и просторно-временских података за које је потребно предикцијом одредити класне вредности. Овакви скупови података се користе у случајевима када након предикције остварености услова за појаву болести није потребно даље чување полазних вредности у оквиру инстанци података тренинг и тест скупа података. У наредном коришћењу софтверског решења корисник креира нове скупове података. Било да су тренинг и тест скуп података креирани ручним уносом података или одабиром датих вредности из базе података, операције које следе јесу операције обуке и креирања предикционог модела, као и само коришћење обученог модела у сврху реализације процеса предикције класних вредости унапред дефинисаних атрибута. Као што је раније наведено, а како се може видети са секвенцијалног дијаграма, резултати предикције се приказују кориснику софтверског система. Након приказа предикционих резултата кориснику је дата могућност да исте искористи у циљу генерисања поруке коју ће проследити одабраној групи клијената. Издвајање циљне групе корисника је омогућено извршењем одговарајучих операција почевши од приступа бази података, преко филтрирања корисника на основу локације најближе метеоролошке станице, па све до селектовања појединачних корисника којима ће се проследити генерисана порука.

Ажурирање базе података, а самим тим и тренинг и тест скупова података, праћено је низом адекватних операција и захтева интеракцију корисника система. Уколико се ради о случају коришћења софтверског система повезаног са метеоролошком станицом или системом метеоролошких станица, интеракција корисника система се заснива на иницирању конекције између базног рачунара и метеоролошке станице и слању захтева за подацима које чине измерене вредности посматраних метеоролошких параметара.

9.2 Архитектура креираног софтверског система

Креирано софтверско решење представља спој већег броја технологија у циљу реализације својеврсног система намењеног употреби од стране крајњих корисника. Имплементација *GreenLife* софтверског система, са једне стране захтева креирање сервиса за остваривање комуникације и преноса пакета података употребом бежичних комуникационих система, док се са друге стране прикупљени подаци обрађују употребом различити data mining техника. Логичка организација креираног софтверског система је представљена на слици 67. Први и у исто време почетни ниво у оквиру креиране логичке архитектуре система представља ниво прикупљања података.



Слика 67: Логичка организација GreenLife *софтверског система*

У оквиру првог нивоа се могу разликовати два начина за прикупљање података. Ова два начина, иако различита, могу се сматрати равноправним начинима за прикупљање и унос података у креирану базу података. Различитост наведених начина за прикупљање података, као и различитост начина уноса података у базу података, огледа се у примењим технологијама.

Први начин прикупљања података представља симбиозу креираног софтверског решења и система аутоматских метеоролошких станица. Посматрано са аспекта реализације комуникације између креираног софтверског решења и метеоролошких станица, потребно је нагласити да се под системом метеоролошких станица може сматрати једна или више аутоматских метеоролошких станица. Процес комуникације и пакетног преноса података истоветан је било да се комуникација обавља са једном или са више метеоролошких станица. Уколико се ради о систему од више метеоролошких станица, успостава комуникације и пакетног преноса података се обавља унапред дефинисаним редоследом (једна метеоролошка станица за другом) како би се избегле колизије међу подацима. Процес успоставе бежичне комуникационе везе иницира корисник система слањем захтева за подацима, након чега на основу послатог захтева следи пријем пакета података, као што је приказано на слици 62, а која илуструје случај коришћења *комуникација са метеоролошком станицом*. Предност првог начина прикупљања и уноса података у оквиру базе података се огледа у томе да корисник система не мора бити присутан на локацији на којој се налази метеоролошка станица како би податке очитао и исте унео у систем.

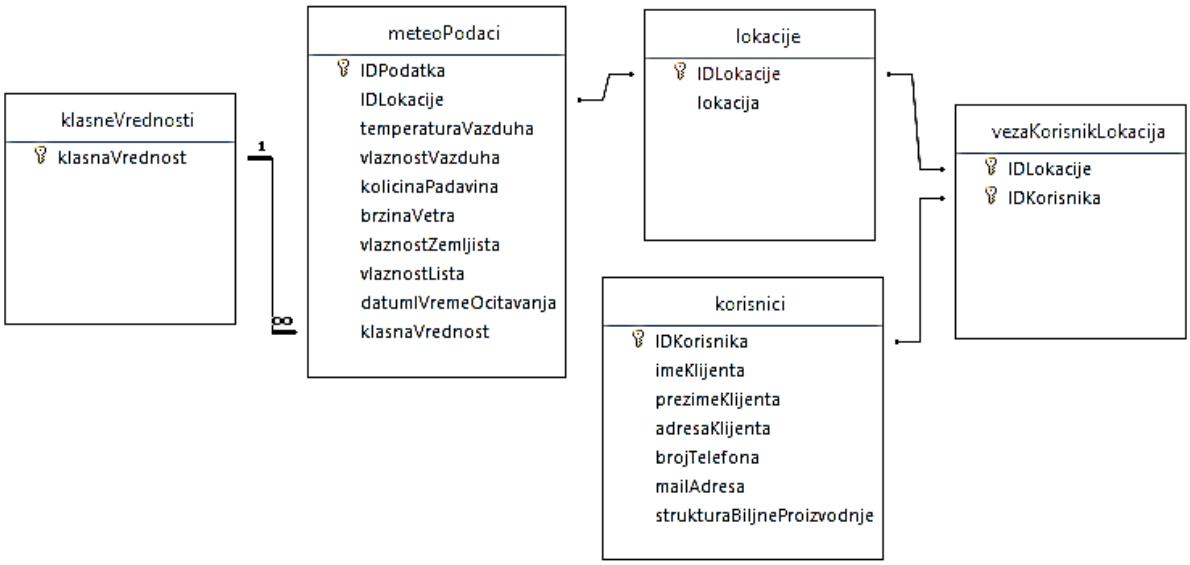
Такође, овакав вид прикупљања података је погодан за коришћење од стране институција или појединаца које у оквиру система имају креирану бежичну мрежу са више метеоролошких станица. Коришћењем оваквог начина уноса података са повећањем броја мерења у току дана повећа се и скуп података са којима *GreenLife* софтверски система оперира, те се повећава и тачност података потребних за креирање и обуку предикционог модела.

Други начин прикупљања података представља ручни унос података од стране корисника система. Имплементација ручног уноса података се огледа у креираним формама у оквиру којих корисник система може иницијално креирати потребне скупове података или унети податке у оквиру креиране базе података. Свако наредно ажурирање базе података представља процес у коме корисник система коришћењем функционалности имплементираних у оквиру форми врши унос нових података.

Овај део система је намењен корисницима система који мерење метеоролошких параметара врше помоћу метеоролошких станица које нису дигитализоване, па захтевају интеракцију корисника у виду очитавања мерних вредности и унос истих у базу података. Корисник система у оквиру овог дела система може унети вредности измерених метеоролошких параметара које је добио од стране Републичког хидрометеоролошког завода или неке друге организације која поседује метеоролошку станицу. Притом, у највећем броју оваквих случајева подаци се добијају у форми папирног извештаја, па је њихов ручни унос у базу података неминован. Имплементација потребних функционалности у оквиру форми за унос података вршена је коришћењем *C#* програмског језика и *WPF* подсистема за рендеровање *GUI*-ја. Коришћењем наведеног подсистема за рендеровање у оквиру *C#* програмског језика, креиран је једноставан и интуитиван графички кориснички интефејс. Овако креиран интерфејс је прилагођен корисницима различитог нивоа обучености и познавања рада на рачунару.

Други ниво логичке организације *GreenLife* софтверског система представља база података. База података се може посматрати као средишњи слој или слој посредник између нивоа за унос података и нивоа за обраду унетих података. Шема креиране базе података је приказана на слици 68. Као што се са дате шеме може видети, креирана база података се састоји од пет међусобно повезаних табела са подацима. Организација података у оквиру сваке од табела са подацима, као и њихова међусобна повезаност, омогућавају адекватно чување података и рад са овим подацима. Табела *meteoPodaci* у оквиру базе података садржи мерења свих метеоролошких параметара и вредности просторно-временских параметара за конкретно мерење. Поред вредности метеоролошких параметара, у оквиру ове табеле се налазе и атрибути који представљају спољашње кључеве и у исто време везу са другим табелама података. Једна од веза је остварена са табелом података *lokacije* у оквиру које се чувају информације о локацијама на којима се налазе метеорлошке станице, док је друга од веза остварена са табелом података *klasneVrednosti*, у оквиру које се налази атрибут који може имати једну од четири класних вредности.

Веза између табеле података *meteoPodaci* и табеле података *klasneVrednosti* је у односу више према један, што значи да више инстанци података у оквиру табеле података *meteoPodaci* узима искључиво по једну вредност атрибута *klasnaVrednost* који се налази у табели података *klasneVrednosti*.



Слика 68: Шема базе података GreenLife софтверског система

Овако остваreном везом, приликом сваког мерења или уноса података у базу податка, свакој од инстанци метеоролошких података се придодаје вредност атрибута који представља информацију о локацији метеоролошке станице са које су добијене дате вредности параметара, као и вредности атрибута који представља информацију о класној вредности. Атрибут *lokacija* у оквиру табеле података *lokacije* креиран је као просторни тип података. На овакав начин атрибут просторног типа носи информацију о географској ширини, географској дужини и надморској висини простора на коме се налази метеоролошка станица. Креирање табеле података у оквиру које се налазе просторни подаци, као и веза ове табеле података са табелом података *korisnici,* условила је креирање целокупне базе података као просторне базе података (енг. *Spatial database*).

Просторна база података представља базу података оптимизовану за чување и рад са подацима који представљају објекте дефинисане у геометриском простору. Већина просторних база података омогућава репрезентацију простих геометријских облика, као што су тачке у простору, линије и пологони.

Неке од просторних база података оперирају са сложенијим структурама попут 3D објеката, линеарних мрежа и различитих тополошких мапа. За разлику од класичних база података које су развијене како би се управљало различитим нумеричким или текстуалним типовима података, просторне базе података захтевају додатне функционалности како би се ефективно управљало просторним подацима. Основне функционалности које карактеришу просторне базе података се огледају у коришћењу геометријских или описних типова података. Такође, просторне базе података се карактеришу коришћењем просторних индекса који се користе како би се убрзале операције над подацима у оквиру базе података. Уобичајени начини индексирања применљиви над подацима у већини база података нису применљиви за коришћење над подацима у оквиру просторних база података. Просторни индекси се користе како би се креирали и оптимизовали просторни упити.

Креирање и коришћење просторне базе података у оквиру *GreenLife* софтверског система иницирано је потребном за повезивањем табеле података *korisnici* са табелом података *lokacije*. С обзиром на чињеницу да табела података *lokacije* садржи просторни атрибут који носи просторне информације, њено повезивање са табелом података *korisnici* било је могуће само креирањем заједничког идентификатора записа. У оквиру базе података је креирана нова табела која за сваку инстанцу података садржи примарни кључ конкретног корисника и примарни кључ локације. На основу вредности ових кључева се креира композитни кључ који представља везу између поменуте две табеле. Креирањем просторне базе података, а самим тим и просторних веза између табела у бази података, омогућено је креирање поменутих просторних индекса и њихово коришћење. Предност овако креиране просторне базе података се огледа у томе да се за сваког корисника у моменту предикције времена хемијског третмана може одредити најближа метеоролошка станица. Применом оваквог метода је осигурано да се предикција времена хемијских третмана врши коришћењем вредности метеоролошких и просторно-временских параметара добијених са најближе метеорлошке станице.

Увођење нове метеоролошке станице или измена локација постојећих метеоролошких станица не захтевају ажурирање листе корисника чије су производне површине покривене конкретним метеоролошким станицама. Просторна удаљеност метеоролошке станице од конкретног корисника је значајна у погледу тачности измерених вредности метеоролошких параметара. Тачност измерених вредности метеоролошких параметара се повећава са смањењем удаљености производних површина корисника од метеоролошке станице.

Табела података *korisnici*, као што се може видети са шеме базе података дате на слици 68, садржи основне информације о корисницима. Уколико се ради о индивидуалном кориснику *GreenLife* система, табела података *korisnici* садржаће само једну инстанцу података у оквиру које ће бити унете основне информације о самом кориснику. Уколико је корисник система институција, табела података *korisnici* садржаће податке о свим клијентима са којима дата институција сарађује. Кључни атрибути ове табеле су *adresaKlijenta*, на основу које се одређује просторна удаљеност од метеоролошких станица, као и контакт подаци клијента које представљају атрибути *brojTelefona* и *mailAdresa* који се користе за слање обавештења о испуњености услова за појаву болести. Атрибут *strukturaBiljneProizvodnje* представља текстуални атрибут чија садржина представља опис пољопривредних активности којима се клијент бави. У оквиру овог атрубута је описано које све биљне врсте клијент узгаја на својим производним површинама. Коришћењем наведеног атрибута у процесу слања обавештења клијентима се може извршити селекција само оних клијената који узгајају биљну врсту за коју је вршена предикција времена наредног хемијског третмана. На овакав начин се поруке шаљу само оним клијентима којима је обавештење о испуњености услова за инфекцију од значаја.

Описана база података *GreenLife* софтверског система је креирана употребом *SpatiaLite* екстензије *SQLite* уграђеног система за управљање базама података. SQLite је у основи садржан у релативно малој *C* програмској библиотеци, у оквиру које је извршена имплементација малог, брзог, самоодрживог, високопоузданог и комплетно описаног *SQL engin*-а за базе података.

Додатна погодност *SQLite-a* се односи на употребу изворног кода који је јавно доступан и може се користити у било које сврхе. За разлику од других *SQL* база података, *SQLite* нема засебан сервер процес. SQLite је компактна библиотека, што значи да са свим укљученим карактеристикама величина библиотеке може бити мања од неколико стотина килобајта, а што зависи од конкретне платформе и оптимизационих подешавања компајлера. Брзина рада *SQLite*-а се у основи повећава са већом доделом меморије. Међутим, предност употребе ове библиотеке се огледа у томе да су перформансе рада најчешће веома добре чак и на машинама са мало меморије. Примери показују да у зависности начина употребе *SQLite* може бити бржи од директног приступа фајл систму. Примена у конкретним апликацијама се огледа у томе да апликација користи *SQLite* функционалности путем једноставних функционалних позива. Коришћењем оваквих позива се смањује латенција у приступу бази података, с обзиром да су функционални позиви у оквиру једног процеса ефикаснији од међупроцесне комуникације. Комплетна база података, почевши од дефиниција, преко табела, индекса, па све до самих података, креира се као један међуплатформски фајл на машини корисника.

Библиотека *SQLite* уграђеног система садржи подршку за просторне податаке у виду имплементације индексирањем путем *R*-стабала и геометријских типова података. Међутим, за сложеније операције и креирање просторних упита оваква подршка за рад са просторним подацима није довољна. Управо из овог разлога, за реализацију базе података *GreenLife* софтверског система коришћена је јавно доступна *SpatiaLite* библиотека која представља проширење *SQLite* језгра у домену пружања подршке и остварења просторних *SQL* могућности. *SpatiaLite* је једноставно интегрисан у *SQLite* како би обезбедио креирање потпуног и моћног просторног *DBMS*-а. *SpatiaLite* екстензија је слична *PostGIS-u, Oracle Spatial-u* и *SQL Serveru* у раду са геопросторним подацима. Како *SpatiaLite* представљаекстензију *SQLite*-а, њихова комбинација није базирана на клијент-сервер архитектури, као што је то био случај и са *SQLite*-ом, већ је за потребе њиховог коришћења усвојена једноставнија персонална архитектура. *SpatiaLite* врши проширење постојеће *SQLite* подршке у циљу остваривања покривености *OGC SFS* спецификације.

SpatiaLite је првенствено оријентисан за коришћење на *Linux* и *Windows* платформама и поред основне библиотеке садржи и помоћне алате који су укључени у основну библиотеку. Додатни алати обухватају рад путем командне линије, графички кориснички интерфејс за рад са просторним базама података и једноставни *GIS* алат за прегледавање и одабир података. Предност употребе *SQLite/SpatiaLite* базе података, поред оперативности са просторним подацима, огледа се и у томе што је целокупни *SQL engine* директно уграђен у саму апликацију. На овакав начин целокупна база података представља обичан фајл који се слободно може копирати, мењати, обрисати или премештати са једне машине на другу, независно од оперативног система на датим машинама. Могућност дистрибуије базе података заједно са апликацијом омогућава несметану инсталацију и коришћење креираног *GreenLife* софтверског система у оквиру кога је уграђена *SQLite/SpatiaLite* база података. Кроз овакву реализацију базе података су елиминисани потенцијални проблеми приликом коришћења креираног софтверског система у виду миграције на другу машину или мигриције на други оперативни систем. С обзиром на чињеницу да се база података дистрибуира у виду фајла и да њена употреба није заснована на клијент-сервер архитектури, корисници креираног софтверског система на својим машинама не морају имати инсталирану серверску подршку.

Трећи ниво у оквиру архитектуре креираног *GreenLife* софтверског система, као што се може видети са слике 67, представља такозвано навођено процесирање или навођену обраду (енг. *Guided processing*). Архитектурни ниво навођеног процесирања обухвата функционалности имплементиране са циљем реализације најбитнијег дела креираног *GreenLife* софтверског система. Сам термин навођено процесирање се односи на чињеницу да се у оквиру овог дела софтверског решења врши обрада података за коју је потребно активно учешће самог корисника система. Активно учешће корисника означава да су резултати обраде података у одређеној мери зависни од подешавања за које је задужен корисник система. Корисник система покреће акције које се односе на обраду података прибављених из базе података, тумачи добијене резултате обраде и доноси одлуке о даљим акцијама.

Логичка организација архитектурног нивоа навођеног процесирања се састоји од три међусобно повезана блока у којима се врши унапред дефинисана обрада. Сваки од блокова обраде се карактерише конкретним функционалностима које се извршавају над излазом који генерише претходни блок обраде овог нивоа. На улаз првог блока обраде навођеног процесирања се доводе подаци над којима се у оквиру овог дела врши конкретна обрада. Добијени резултати обраде се даље прослеђују средишњем блоку овог нивоа у оквиру кога се извршава даља обрада. Резултати спроведене обраде, у оквиру другог блока навођеног процесирања, користе се од стане трећег и у исто време последњег блока обраде овог нивоа креираног софтверског решења. Сваки од наведених блокова се састоји од низа операција задужених за конкретну обраду.

Први блок обраде под назвом *обука модела* обухвата активности које се извршавају са циљем обуке предикционог модела коришћењем тренинг скупа података са познатим исходом. Шематска организација блока *обука предикционог модела* приказана је на слици 69. Обука придикционог модела почиње селекцијом података из базе података у циљу креирања тренинг скупа података. Корисник најпре у оквиру почетне форме врши селекцију параметара потребних за креирање упита на основу којих ће се селектовати подаци у оквиру базе података. Након добијања података из базе података, корисник врши одабир жељених инстанци података и креира тренинг скуп података са познатим исходом. Такође, у овом кораку корисник коришћењем имплементираних функционалности може одабрати већ креирани тренинг скуп података са познатим исходом над којим се могу применити различити предикциони алгоритми. Процес обуке предикционих модела је омогућен корисницима софтверског система кроз примену већег броја предикционих алгоритама. Корисник система селекцијом жељених предикционих модела започиње процес обуке. Имплементацијом функционалности везаних за процес обуке предикционих модела дефинисано је да се сваки од селектованих предикционих алгоритама над скупом података са познатим исходом извршава у оквиру једне нити. Овакво извршење значи да се за сваки од примењених предикционих алгоритама креира се по једна нит.

Слика 69: Шематска организација блока обука модела навођеног процесирања

Као резултат примене *n* предикционих алгоритама над јединственим тренинг скупом података са познатим исходом добија се *n* обучених предикционих модела. Сваки од креираних предикционих модела пролази процес евалуације. Процес евалуације је креиран јединствено за све имплементиране предикционе алгоритме. Задатак евалуације јесте поређење перформанси и одређивање степена тачности обучених предикционих модела. На основу урађене евалуације корисник система врши одабир једног модела који ће се користити у наредним блоковима обраде. Како корисник система, као што је раније наведено, може бити лице са основним познавањем рада на рачунару, резултати евалуације самом кориснику презентују се описно.

Како се за сваки од модела евалуирају истоветни параметри, врши се поређење датих параметара и кориснику система презентује који од модела је најпогоднији за коришћење. Описна информација о томе који је најбоље обучени предикциони модел садржи и податак о процентуалној тачности конкретног модела, као и податак који показује колико је тачност најбоље рангираног предикционог модела већа од тачности осталих обучених предикционих модела. Кориснику система је, и поред извршене евалуације, остављена могућност одабира једног од обучених модела, било да је то модел са најбољим перформансама или било који други предикциони модел из групе претходно обучених.

Предност имплементације овако конципираног блока обраде у оквиру нивоа навођеног процесирања се огледа у томе да овај блок обраде омогућава кориснику креираног софтверског система могућност вршења обуке сопствених предикционих модела заснованих на скупу података са познатим исходом који је корисник сам креирао. На овакав начин је кориснику софтверског система омогућено да поред креираних и уграђених предикционих модела изврши креирање модела на основу новог или модификованог скупа података. Свака измена над подацима у тренинг скупу, допуна тренинг скупа или креирање новог тренинг скупа кориснику система даје могућност покретања обуке и креирања новог предикционог модела. Новокреирани предикциони модел потенцијално може водити бољој тачности будуће предикције. Кроз процес уноса података и креирања новог тренинг скупа података са познатим исходом корисник система може дефинисати, унети и користити вредности параметара специфичних за услове узгоја гајене биљке чиме је кориснику дата могућност додатног специфицирања параметара на којима ће се заснивати обука предикционог модела, а самим тим и његово касније коришћење у процесу предикције.

Излаз блока *обука модела* је обучени предикциони модел који је одабран од стране корисника, а на основу добијене повратне информације из процеса евалуације. Одабрани предикциони модел представља улаз у блок обраде под називом *креирање модела*. Шематска организација блока *креирање модела* у оквиру навођеног процесирања је дата на слици 70. Имплементиране функционалности у оквиру овог блока омогућавају кориснику додатно подешавање претходно обученог модела. Уколико је додатно подешавање обученог модела у зависности од примењеног предикционог алгоритма могуће, кориснику креираног софтверског система се динамички у оквиру корисничке форме приказују опције помоћу којих се обучени модел може додатно подесити. На пример, уколико је корисник у процесу обуке одабрао обучени модел заснован на стаблима одлучивања, у блоку *креирања модела* корисник може извршити одсецање (енг. *pruning*) појединих делова стабла. Одсецање делова стабла одлучивања подразумева смањивање његове величине у циљу оптимизације стабла одлучивања.

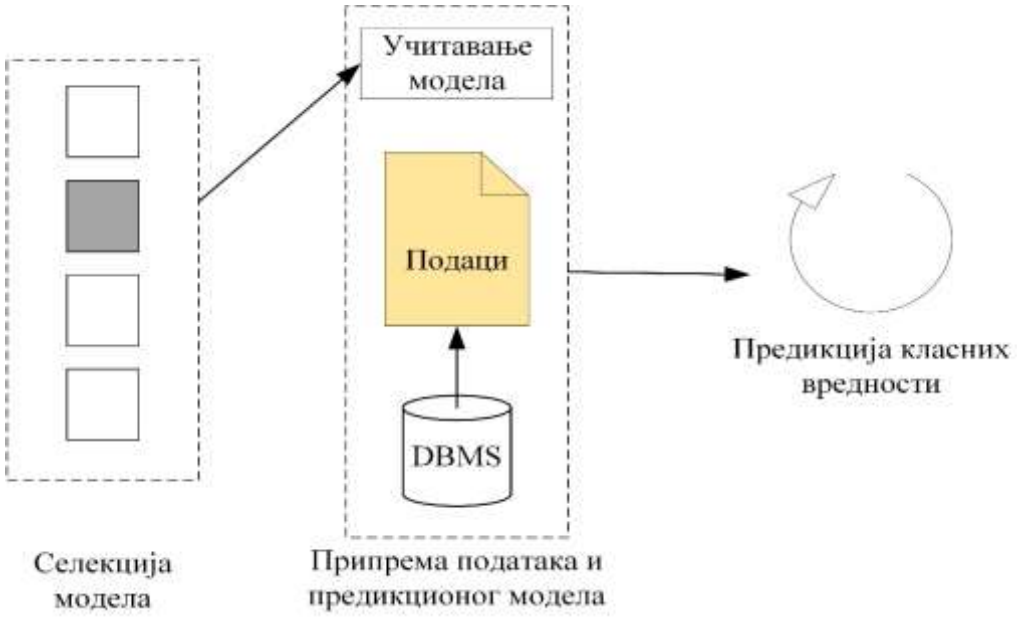
Са једне стране, велико стабло одлучивања може бити креирано тако да представља искључиво тренинг скуп података над којим је креирано, док се у оквиру оваквог стабла одлучивања нове инстанце података јако тешко уклапају. Са друге стране, мало стабло одлучивања најчешће не представља тренинг скуп података на прави начин, па самим тим и предикција нових вредности не може бити веродостојна. Управо из ових разлога се кориснику у овом кораку нуди могућност одсецања појединх делова стабла одлучивања чиме се проналази баланс између сувише великог и сувише малог стабла одлучивања. Одсецањем се најпре елиминишу гране стабла које су најудаљеније од корена стабла, чиме се смањује ниво стабла. Након додатних подешавања обученог предикционог модела врши се чување предикционог модела. Овако креирани предикциони модел сматра се потпуним моделом који корисник креираног софтверског система користи у процесу предикције остварености услова за појаву болести, а самим тим и предикцију времена хемијских третмана.

Слика 70: Шематска организација блока креирање модела навођеног процесирања

Понављањем процеса обуке и креирања модела коришћењем предикционих алгоритама над различитим скуповима података са познатим исходом, корисник система може креирати већи број предикционих модела који ће се користити за предикцију времена хемијских третмана за различите гајене биљке.

Трећи дефинисани блок обраде у оквиру навођеног процесирања обухвата функционалности којима је регулисано коришћење креираних предикционих модела. Шематска организација овог блока обраде дата је на слици 71. Као што је био случај и код претходних блокова обраде и код овог блока је дефинисан очекивани улаз и очекивани излаз.

Посматрано са становишта улаза у оквиру овог блока, могу се дефинисати два начина обезбеђења адекватног улаза. Први од начина заправо је наставак реализације претходног блока где корисник практично након креирања и чувања модела прелази на његову практичну употребу. Други начин представља ситуацију у којој корисник у оквиру овог блока може извршити селекцију једног од раније креираних предикционих модела. Заједничко за оба начина коришћења, као што се може видети, јесте чињеница да улаз у овај блок свакако мора бити креирани предикциони модел. Након одабира жељеног предикционог модела корисник креираног софтверског система мора креирати скуп података над којима ће се вршити предикција. Као што је то био случај са креирањем тренинг скупа података, и у овом кораку корисник врши селекцију жељених инстанци података у оквиру базе података. Након селектовања жељених инстанци података и њиховог организовања у адекватан скуп података, поред одабира предикционог модела, испуњен је и други предуслов за коришћење одабаног предикционог модела.



Слика 71: Шематска организација блока предикција навођеног процесирања

Коришћење одабраног предикционог модела огледа се у његовој примени над креираним скупом података како би се предикцијом одредиле вредности зависне променљиве. Процес предикције је поновљив, као што се може видети на слици 71, што значи да се једним одабраним предикционим моделом може извршити предикција над већим бројем скупова података.

То значи да се, уколико се ради о кориснику који добија податке са више локација, за сваку од локација може креирати засебан скуп података, па се за сваки од креираних скупова података може обавити засебан процес предикције. Предикција организована на овакав начин требало би да буде већег степена тачности, с обзиром на чињеницу да се предикција извршава над подацима организованим према припадности локалитетима, а самим тим и према томе са које су метеоролошке станице добијени.

Потребно је нагласити да се раније наведена међусобна веза дефинисана између блокова обраде у оквиру навођеног процесирања, као и условљеност преноса података од једног до другог блока остварује у ситуацијама када корисник по први пут креира потребни предикциони модел или када корисник покреће поновно креирање предикционог модела коришћењем новог или модификованог скупа података са познатим исходом. У оваквом случају, као што је детаљно описано прегледом активности у оквиру сваког од наведена три блока корисник креће од креирања скупа података, преко обуке предикционог модела, до његовог креирања и финалног коришћења. У ситуацијама када корисник поседује већ креиране предикционе моделе једноставно може прескочити прва два блока обраде и одмах прећи на трећи блок обраде који се односи на коришћење предикционог модела.

Четврти и уједно последњи ниво архитектуре креираног *GreenLife* софтверског система представља презентациони ниво. Презентациони ниво пружа функционалности потребне како би се обезбедила адекватна интеракција између корисника и креираног софтверског система. Коришћењем функционалности имплементираних у оквиру презентационог нивоа регулисано је управљање креираним софтверским системом од стране корисника, што подразумева унос података, селекцију података, креирање потребних скупова података, креирање предикционих модела, предикцију класних вредности и све остале функционалности предвиђене описаним случајевима коришћења. Целокупни презентациони ниво је креиран тако да буде у складу са *HCI* (енг. *Human Computer Interaction*) стандардом.

Функционалности презентационог нивоа у оквиру архитектуре креираног софтверског система, поред обезбеђења механизама за управљање системом од стране корисника, имплементиране су тако да целокупно окружење буде интуитивно и пријатно за коришћење. Интуитивност креираног *GUI*-а је посебно значајна у погледу коришћења креираног софтверског система од стране корисника који нису вешти у коришћењу рачунарског хардвера и софтвера. Презентациони ниво се, као што се може видети на слици 67, састоји од два узајамно равноправна и повезана блока. Први блок је блок презентације у оквиру кога корисник интерагује са креираним софтверским системом. У оквиру овог блока се кориснику приказују резултати предикције спроведене над раније креираним скупом података. Такође, у оквиру овог блока корисник може оперирати инстанцама података у оквиру скупа података, додати рецимо нове инстанце података, изменити вредности података у постојећим инстанцама и поновно покренути процес предикције.

Презентовани резултати извршене предикције у оквиру овог блока могу представљати финалну фазу коришћења креираног софтверског система, уколико се ради о коришћењу система од стране индивидуалног корисника. Други блок у оквиру нивоа презентације представља групу функционалности којима је обезбеђена комуникација са клијентима. Уколико је корисник креираног софтверског система институција или појединац, након извршене предикције а помоћу импелементираних функционалности, они могу послати обавештење клијентима о правом времену хемијског третмана. Посматрано из угла корисника система, функционалности другог блока презентационог нивоа се покрећу након извршене предикције и презентовања резултата у оквиру *GUI* окружења кога чине функционалности блока презентације. Добијене резултате предикције корисник система може преточити у текстуално обавештење које ће проследити одабраној групи клијента. Такође, корисник система у оквиру обавештења може укључити и саме резултате предикције. У оквиру овог блока корисник система врши селекцију оних клијената којима ће конкретно обавештење бити прослеђено. Селекција клијената је заснована на два основна параметра. Први параметар је адреса клијента. На основу адресе клијента се врши одређивање најближе метеоролошке станице.

Уколико је клијенту најближа метеоролошка станица управо она са које су добијени подаци коришћени у предикцији, дати клијент је кандидат за слање обавештења. Други параметар представљају инфомације уписане у бази података у табели корисник у оквиру атрибута *struktura biljne proizvodnje*. Ове информације се користе у процесу провере којом се утврђује да ли конкретном клијенту може бити од интереса генерисано обавештење, у зависности од тога које биљне врсте узгаја на својим производним површинама. Уколико клијент у структури биљне производње има уписане биљне врсте које су обухваћене предикцијом, клијенту се прослеђује генерисано обавештење. Уколико провера структуре биљне производње покаже да клијент на својим производним површинама не узгаја гајене биљке, обухваћене предикцијом у том случају, генерисано обавештење њему није од значаја, те му се и не прослеђује.

Имплементација *GreenLife* софтверског система, заснованог на архитектури представљеној на слици 67, је извршена коришћењем више технологија. Као што је раније наведено, графички кориснички интерфејс је дизајниран коришћењем *WPF* подсистема за рендеровање, док је сама имплементација функционалности интерфејса извршена коришћењем *C#* програмског језика. Како функционалности интерфејса подразумевају обезбеђивање активности интеракције корисника са креираним софтверским системом, позадинске функционалности, независне од самог корисника креираног софтверског система, су имплементиране коришћењем *C#* програмског језика. На овакав начин је целокупна имплементација обједињена у оквиру једног пројекта. Ради лакше организације самог софтверског система, као и имплементације више међусобно повезаних слојева, имплементација овог система је заснована на коришћењу шаблона. Основни примењени шаблон који међусобно повезује све нивое архитектуре креираног софтверског система јесте *Layer* шаблон. Како база података представља засебан фајл, креиран коришћењем *SQLite/SpatiaLite,* било је потребно омогућити извршење свих потребних операција над базом података употребом контрола у оквиру креираног *GUI*-а. Управо из тог разлога је имплементација комуникације са базом података на свим нивоима извршена коришћењем *C#* програмског језика.

На овакав начин је елиминисано коришћење додатних конвертора или увођење нових технологија, чиме се задржала полазна једноставност у раду са базом података. Централни слој креираног софтверског система представљају функционалности које одговарају архитектурном нивоу навођеног процесирања. У оквиру овог слоја, коришћењем *C#* програмског језика, извршена је имплементација одређене групе предикционих алгоритама. Међутим, како је већи број раније описаних и у склопу прототипа апликације тестираних data mining предикционих алгоритама доступан у оквиру *Weka* data mining алата било је потребно инкорпорирати *Weka* функционалности у већ креирани *C#* пројекат. *Weka* data mining алат је, заједно са свим функционалностима које нуди, имплементиран у *Java* програмском језику. Како би се обезбедила компатибилност између пројекта креираног у *C#* програмском језику и функционалности *Weka* алата написаних у *Java* програмском језику било је потребно превести *weka.jar* у *weka.dll* фајл. Процес превођења *weka.jar* фајла у *weka.dll* фајл је извршен коришћењем *IKVM.NET* алата. Овај алат представља имплементацију *Java*-е за *Microsoft .NET Framework*. Практично *IKVM.NET* укључује *Java* виртуалну машину имплементирану у *.NET*-у, *.NET* имплементацију *Java* библиотека класа, као и алат који омогућава *Java* и *.NET* интероперабилност. Коришћење овог алата омогућава коришћење *Java* библиотека у оквиру *.NET* апликација. *IKVM* алат садржи *ikvmc* компајлер чијом се употребом *Java* библиотеке могу конвертовати у *.dll* фајлове, чиме је омогућено њихово коришћење у оквиру *.NET* апликација. Како *core Weka data mining alata* представља заправо *java* библиотеку у ознаци *weka.jar,* употребом *ikvmc* и његовим позивањем у оквиру командне линије, извршено је конвертовање *weka.jar* библиотеке у *weka.dll*. На овакав начин је омогућено укључивање *weka.dll* фајла као сваког другог *.dll* фајла у оквиру *C#* пројекта. Укључивањем *weka.dll* фајла у *C#* пројекат, функционалности *weka* data mining алата су постале доступне за коришћење из *.NET* окружења. Приликом имплементације функционалности везаних за коришћење data mining алгоритама садржаних у оквиру *weka.dll*-а, унутар креираних *C#* функција је вршено позивање *weka* класа и метода имплементираних у оквиру ових класа.

Обрађивање резултата добијених извршењем weka метода, као и прослеђивање одговарајућих параметара, имплементирано је коришћењем *C#* програмског језика. Овако организованом имплементацијом је у основи добијен спој два различита програмска језика, чиме су успешно обједињене погодности *weka* data mining алата са погодностима које *C#* програмски језик нуди у погледу креирања *GUI* окружења и обезбеђења адекватне интеракције корисника и система.

Имплементација фунционалности *GreenLife* софтверског система задужених за процес комуникације и слања обавештења клијентима заснована је на примени *Brocker* шаблона. Имплементацијом датих функционалности је обезбеђено слање порука коришћењем једног од два сервиса. Први од сервиса се заснива на слању обавештења у виду *SMS* поруке, док се употребом другог сервиса обавештење може проследити помоћу *email*-а. У случају коришћења *SMS*-а као начина за слање обавештења, на страни корисника система се захтева коришћење *GSM/GPSR* модема, а помоћу кога се обавља успостава комуникације и пренос пакета података који одговара садржини *SMS* поруке. Уколико се као сервис за слање обавештења користи *email*, није потребно коришење додатног хардвера. Међутим, коришћење *email*-а као сервиса за слање обавештења изискује мрежно повезивање корисника креираног софтверског система. Употреба било једног било другог сервиса омогућује задовољавајуће перформансе у погледу размене порука.

10. Резултати истраживања и дискусија

Хемијска заштита гајених биљака представља један од најсложенијих процеса у поступку узгоја и култивације гајених биљака. Климатске промене и комерцијални узгој гајених биљака на великим површинама су допринели развоју великог броја патогена. Ни један од патогена се не може посматрати као нови патоген настао под дејством климатских промена, с обзиром на чињеницу да је највећи број познатих патогена регистрован јако давно. Утицај климатских промена се огледа у томе да тренутни метеоролошки услови све више погодују развоју и распрострањености разних биљних патогена. Механизми заштите гајених биљака су различити, у зависности од начина узгоја самих биљака. Најраспрострањенији механизам заштите, применљив у комерцијалној производњи, јесте хемијска заштита. Успешност хемијске заштите зависи од већег броја фактора: временског тренутка у коме је обављена хемијска заштита, одабира адекватног хемијског препарата, концентрације хемијског препарата и начина примене. Несумљиво највећи утицај на успешност хемијске заштите има временски тренутак у коме је заштита изведена. Временски тренутак хемијске заштите представља временски период у коме се са најмањом концентрацијом хемијског препарата може сузбити појава конкретног патогена. Поред количине хемијског препарата, хемијска заштита изведена у правом тренутку се може обавити и хемијским препаратима мање штетним по људско здравље и околину. Такође, адекватним сузбијањем се може редуковати број хемијских третмана, с обзиром да се за сузбијање патогена не врши понављање процеса хемијске заштите. Међутим, иако представља један од најбитнијих фактора у хемијској заштити гајених биљака, одређивање временског тренутка у коме је најпогодније обавити хемијску заштиту веома је комплексан процес. Сам процес одређивања времена хемијских третмана се заснива на процесу одређивања остварености услова за развој конкретних биљних патогена, а самим тим и остварености услова за појаву инфекције гајених биљака датим патогеном. Управо из ових разлога, комплексност одређивања времена хемијских третмана се огледа у великом броју чинилаца који морају бити обухваћени. Одређени број чинилаца представљају метеоролошких услови и доба године у коме се одређени патоген може јавити.

Правилним сагледавањем метеоролошких услова у датом периоду времена се може извршити одређивање остварености услова за појаву биљних болести, а самим тим и одредити временски тренутак у коме је најпогодније обавити хемијску заштиту.

Развој информационо комуникационих технологија омогућава примену различитих алгоритама и поступака обраде података у циљу одређивања вредности непознате променљиве на основу вредности познатих променљивих. Како се процес одређивања времена хемијских третмана може сагледати као процес у коме је на основу познатих вредности посматраних чинилаца потребно одредити најпогоднији временски тренутак хемијске заштите, решење овако дефинисаног проблема се може пронаћи применом информационо комуникационих технологија. Примена информационо комуникационих технологија у пољопривредној производњи је сваким даном све заступљенија. Могућности које нуде информационо комуникационе технологије су искоришћене у склопу овог истраживања, са циљем креирања софтверског решења које би обухватило процес прикупљања података коришћењем адекватних сензора, њихову дистрибуцију коришћењем бежичних комуникационих система и њихову обраду коришћењем data mining техника. Примена data mining техника се огледа у обради претходно креираних скупова података који представљају везу између различитих чинилаца који утичу на одређивање остварености услова за појаву биљних болести, а самим тим утичу и на одређивање времена хемијских третмана. Пременом одговарајућих data mining алгоритама врши се креирање предикционих модела одређеног степена тачности, чијим се даљим коришћењем може извршити предикција времена хемијских третмана.

Тачност предикционих модела, а самим тим и тачност будуће предикције засноване на креираним предикционим моделима, условљена је процесом обуке предикционих модела. Током процеса обуке предикционих модела кључна су два фактора. Први фактор представља одабрани алгоритам који се користи за обуку предикционог модела, док други фактор представља креирани тренинг скуп података са познатим исходом.

Значајност одабраног алгоритма се огледа у могућностима обраде података у оквиру тренинг скупа и њихове најбоље репрезентације. Утицај одабаног алгоритма се огледа се у томе да ли и у којој мери алгоритам детектује зависност између података у оквиру тренинг скупа у циљу креирања шаблона, као и каснијег креирања предикционих правила базираних на међусобним зависностима између података у тренинг скупу. Како се најчешће ради о тренинг скупу података са познатим исходом, задатак примењеног алгоритма јесте креирање предикционих правила у којима фигурира међусобна веза између зависног атрибута са једне стране и независних атрибута са друге стране. Сваки од независних атрибута има одређени утицај на резултат предикције, односно на вредност зависног атрибута добијену предикцијом. Data mining, као област која се бави обрадом података, садржи велики број алгоритама различитих категорија који се могу успешно применити на решавање проблема предикције. Било да се ради о класификационим алгоритмима, кластеризационим алгоритмима или алгоритмима генерализоване линеарне регресије, сваки од алгоритама може бити примењен на решавање одређене групе проблема.

Одабиру конкретног алгоритма из одређене групе алгоритама претходи процес процене компатибилности датог алгоритма са тренинг скупом података, као и процена о томе да ли конкретни алгоритам може одговорити захтевима будуће предикције. Овакав приступ омогућава одабир алгоритама заснован на карактеристикама конкретних алгоритама и на њиховој предвиђеној намени. Током спроведеног истраживања за реализацију предикције времена хемијских третмана одабране су две групе алгоритма: математички алгоритми генерализоване линеарне регресије и класификациони data mining алгоритми. Ови алгоритми су одабрани на основу компатибилности са креираним скуповима података на којима се базира креирање предикционих модела и на основу очекиваних зависности између података. У оквиру одабраних група алгоритама се налазе алгоритми за које је, на основу њихових карактеристика, постојала претпоставка да могу бити успешно примењени у процесу обуке модела, креирању предикционих правила и процесу предикције.

Сама претпоставка је креирана поређењем карактеристика датих алгоритама и примера њихове раније примене са карактеристикама података у оквиру тренинг скупа подататка и предикционим проблемом који је потребно реализовати. Управо из ових разлога су се, приликом избора између већег броја група алгоритама у склопу истраживања, издовијиле две поменуте групе. Како обе поменуте групе садрже већи број алгоритама, на основу карактеристика сваког од појединачних алгоритама је извршен одабир одређеног броја алгоритама из обеју група. Поред припадности одабраних алгоритама конкретним групама које се одликују заједничким перформансама, сваки од алгоритама се одликује неким специфичним својствима и елементима који утичу на процес обуке предикционог модела, као и на сами процес предикције. Након одабира одређеног броја алгоритама у циљу поређења карактеристика одабраних алгоритма. извршена је евалуација сваког алгоритма појединачно. Процес евалуације алгоритама је спроведен применом двеју метода.

Методом унакрсне валидације, као првом примењеном методом, одабрани алгоритми су валидирани коришћењем података у оквиру тренинг скупа података са познатим исходом. Валидација тачности примењених алгоритама у домену креирања предикционог модела, као и примене креираних предикционих модела у процесу предикције, вршена је у два пролаза са 5 и 10 *fold*-ова креираних у оквиру тренинг скупа података. На овакав начин се унакрсном валидацијом са 5 и 10 *fold*-ова перформансе креираних предикционих модела пореде кроз дефинисани скуп параметара. Неизоставна је чињеница да се у процесу обуке и креирања предикционих модела, као и у процесу унакрсне валидације, користе подаци који припадају тренинг скупу података. Унакрсна валидација креирањем *fold*-ова и процесом понављања поступка обезбеђује дељење тренинг скупа на онолики број делова колико је дефинисано *fold*-ова, чиме се у различитим пролазима у поступку валидације користе различити подскупови тренинг скупа података. Примена унакрсне валидације је омогућила издвајање појединих алгоритама као алгоритама са бољим перформансама у односу на остале одабране алгоритме. Такође, извршење унакрсне валидације за дефинисаних 10 *fold*-ова је обезбедило прецизније резултате процене тачности креираних предикционих алгоритама. Овако дефинисана унакрсна валидација је показала већу тачност предикционих

алгоритама у односу на унакрсну валидацију са 5 *fold*-ова. Како би одредило који од алгоритама показује најбоље перформансе у погледу креирања и коришћења предикционих модела намењених предикцији остарености услова за појаву инфекције, а самим тим и предикцији времена хемијских третмана примењена је још једна метода евалуације.

Друга примењена метода у процесу евалуације не користи податке из тренинг скупа података, већ се процес евалуације базира на коришћењу података из засебног тест скупа података. Као што је раније наведено, а потврђено у већем броју истраживања, процес евалуације алгоритама који се примењују на креирање предикционих модела се најефикасније обавља управо тестирањем заснованим на коришћењу тест скупа података са познатим исходом. Креирани тест скуп података је потребно да по структури и типовима података које садржи одговара тренинг скупу података са познатим исходом, а на коме је заснован креирани предикциони модел. Процена тачности креираних предикционих модела, коришћењем тест скупа података, може се посматрати као практична примена креираних предикционих модела. Једина разлика између процеса тестирања предикционих алгоритама и предикције у реалним ситуацијама се огледа у постојању тачне вредности зависног атрибута за сваку од инстанци података. Процена тачности је била заснована на извршењу процеса предикције за инстанце података у тест скупу података. Предикцијом су добијене вредности зависног атрибута поређене са стварним вредностима доступним у оквиру тест скупа података. Поред самог процеса поређења, вршено је и израчунавање већег броја параметара на основу креираних *confusion* матрица. Израчуната тачност креираних предикционих модела, као и добијени параметри, адекватни су показатељ успешности одабраних предикционих алгоритама. Примењене методе у случају генерализованих линеарних алгоритама, а на основу посматраних параметара, показале су да се Псеудо квадратна дискриминациона анализа може издвојити као најефикаснија метода предикције времена хемијских третмана за посматрани скуп података. Са друге стране, када се ради о класификационим data mining техникама, сви посматрани и поређени алгоритми се одликују сличном тачношћу предикције. Стопа тачности поређених алгоритама се разликује за веома мали проценат.

Међутим, уколико је потребно издвоји најбоље кандидате за примену, издвајају се *J48* и *Random Forest* алгоритам. *J48*, предикциони алгоритам заснован на стаблима одлучивања, се додатно може унапредити процесом одсецања иницијално креираног стабла, чиме се добија бољa тачност овако креираног предикционог модела. Процес унапређења предикционог модела, креираног коришћењем *J48* стабла одлучивања описан је у овом поглављу.

Процес обуке предикционих модела и припрема предикционих модела за будуће коришћење у процесу предикције започиње креирањем адекватних скупова података са познатим исходом. Како се инстанце података у оквиру тренинг скупа података, као што је раније наведено, састоје од вредности независних и зависних атрибута, декларација ових атрибута мора бити јасно наглашена како би се направила разлика између њих. Поред декларације атрибута, посматрано са аспекта обуке предикционог модела, тачност вредности атрибута у оквиру инстанци података тренинг скупа са познатом исходом има великог удела на сами процес обуке. Присутност *outlier*-а у оквиру тренинг скупа података са познатим исходом, на коме се базира креирање предикционог модела, може утицати на тачност креираног предикционог модела, а самим тим и на тачност резултата будуће предикције, засноване на овако креираном предикционом моделу. Управо из ових разлога је процес анализе креираних скупова података у циљу детекције потенцијалних *outlier*-а, као и њиховог отклањања из креираног скупа података. Један од основних процеса који претходе обуци предикционог модела. Анализа скупова података се може извршити применом већег броја метода. Примењене методе су у склопу истраживања показале присуство *outlier*-а у оквиру иницијалног креираног скупа података.

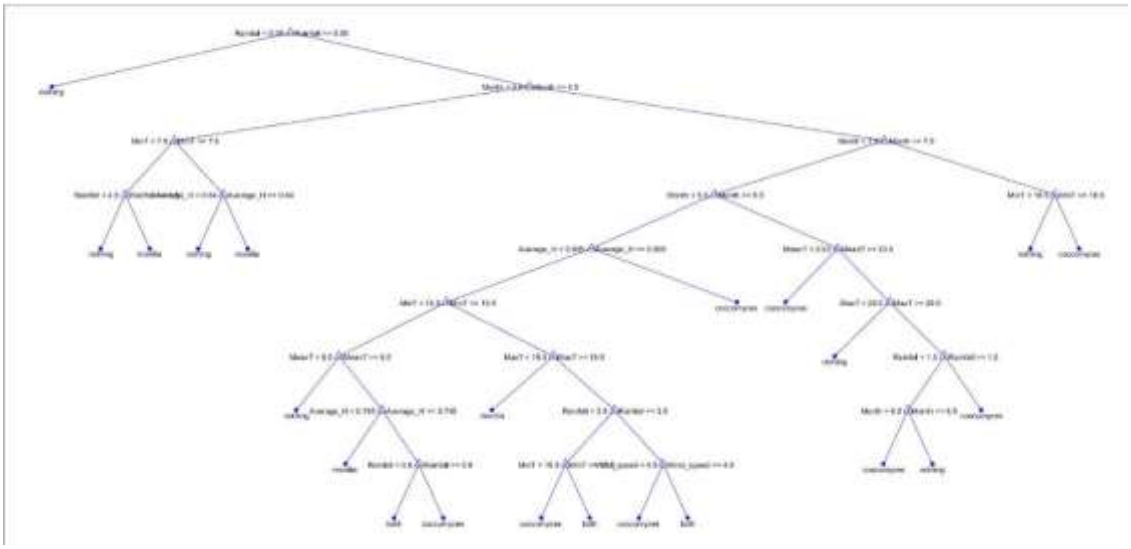
Како је у оквиру креираног скупа података који се састојао од метеоролошких и просторно-временских података, који представљају независне вредности атрибута, као и класних вредности зависног атрибута, детектовано присуство outlier-а, исти су отклоњени применом одговарајућих метода. Најчешће се радило о погрешно унетим вредностима појединих атрибута које су успешно реализоване заменом одговарајућим вредностима. Како је реализација процеса обуке предикционих модела условљена тачношћу података у оквиру тренинг скупа

података, у оквиру спроведеног истраживања је извршена и процена утицаја *outlier*-а на сами процес предикције. Целокупни процес обуке и креирања предикционих модела, као и процена тачности овако креираних предикционих модела, паралелно је обављен коришћењем иницијално креираног тренинг скупа податка (скуп података са *outlier*-има) и тренинг скупа података који је настао након отклањања *outlier*-а. Поређењем перформанси креираних предикционих модела, заснованих на тренинг скуповима података са и без *outlier*-а за предикционе моделе креиране употребом математичких и класификационих алгоритама, уочава се да се предикциони модели креирани коришћењем тренинг скупа података без *outlier*-а одликују већом процентуалном тачношћу.

Утицај *outlier*-а се може огледати у креирању лажно позитивних резултата, што се у случају креирања предикционог модела и његовог тестирања може одразити на добијену тачност. У оваквим случајевима постоји могућност да се након спроведеног процеса процене тачности креираног предикционог модела за тачност креираног предикционог модела добије тачност која је већа од реалне вредности. *Outlier*-и који узрокују овакво понашање предикционог модела теже се детектују од осталих *outlier*-а. Међутим, генерисање лажно позитивних резултата и добијање погрешне вредности за процентуалну тачност креираног предикционог модела, може детектовати тестирање применом новокреираног тест скупа података са познатим исходом, а у оквиру кога све инстанце података са познатим исходом садрже вредности које проверено нису *outlier* вредности. Коришћени тест скуп у оквиру спроведеног истраживања није садржао *outlier* вредност, самим тим добијена тачност креираних предикционих модела одговара реалној тачности.

Након процеса обуке предикционог модела се може спровести процес креирања предикционог модела или, прецизније речено, процес унапређења предикционог модела подешавањем параметара карактеристичних за рад самог предикционог модела. Како су се предикциони модели, креирани применом алгоритама из групе класификационих data mining алгоритама, готово изједначили у погледу тачности креираног модела, анализирано је могуће унапређење предикционог модела креираног коришћењем J48 стабла одлучивања.

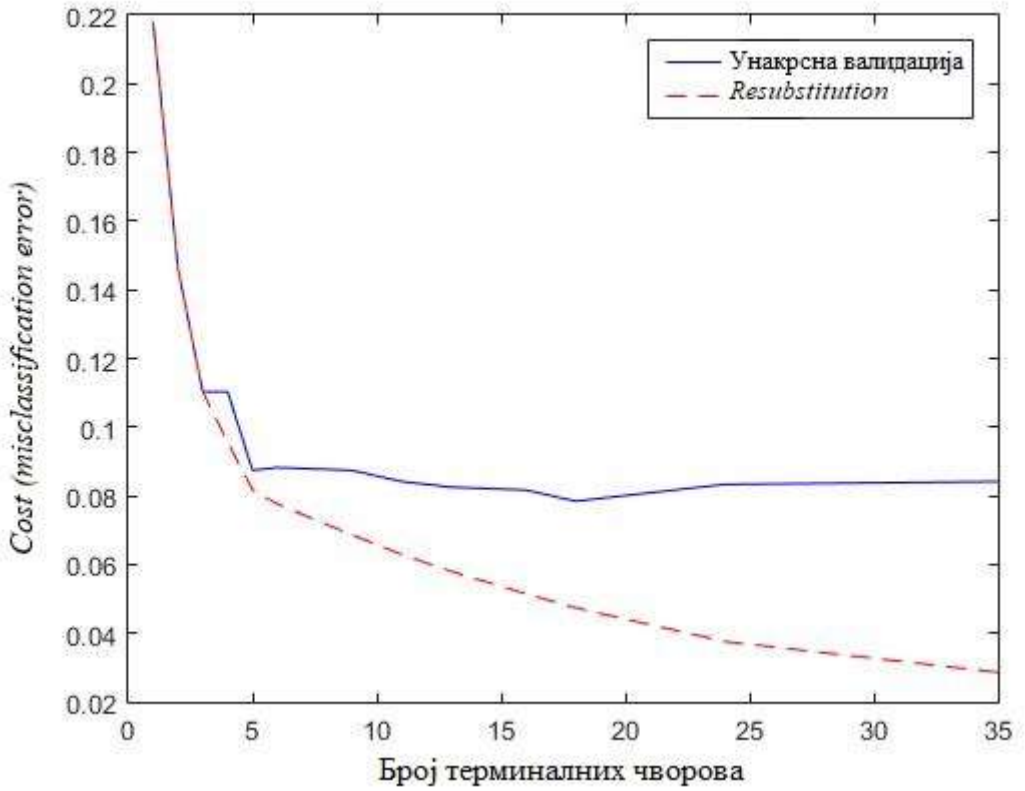
Идеја за унапређење овако креираног предикционог модела, одсецањем креираног стабла одлучивања, потекла је од раније спроведених истраживања у којима је J48 стабло одлучивања показало најбоље резултате приликом коришћења над сличним скуповима података [128]. Предикциона правила иницијално креираног предикциног модела су применом *J48* алгоритма организована у форми стабла одлучивања. Овако креирано стабло одлучивања је приказано на слици 72.



Слика 72: Креирано стабло одлучивања применом J48 алгоритма

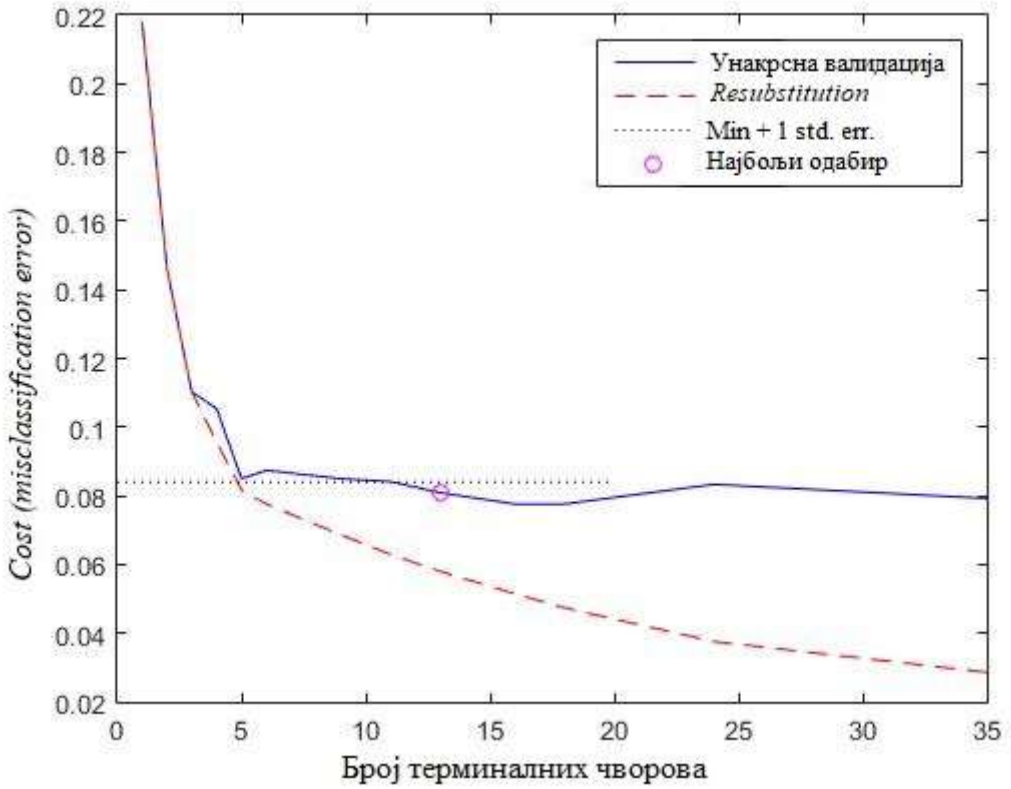
Као што се може видети са добијеног стабла одлучивања, ово стабло одлучивања има десет нивоа. Свака путања у креираном стаблу одлучивања може представљати једно од могућих правила одлучивања. У циљу добијања бољих предикционих резултата, извршен је процес одсецања одређених нивоа у стаблу одлучивања. У циљу одређивања оптималног броја нивоа у оквиру креираног стабла одлучивања, извршено је израчунавање *resubstitution* грешке и грешке унакрсне валидације за 10 *fold*-ова. Израчунавање грешке унакрсне валидације за 10 *fold*-ова је одабрано због чињенице да је конкертни модел најбоље реултате показиво приликом тестирања са 10 *fold*-ова. *Resubstitution* грешка редистрибуције представља однос грешке добијене у оквиру тренинг скупа података. Добијени резултати израчунавања за *resubstitution* грешку износе 0,0286, док за грешку унакрсне валидације износе 0,0817. Поређењем добијених резултата се може закључити да је за конкретно стабло одлучивања процењена грешка унакрсне валидације већа од *resubstitution* грешке.

Овако добијени податак указује да креирано стабло одлучивања класификује оригинални тренинг скуп података веома добро. Међутим, структура креираног стабла одлучивања је осетљива на конретни тренинг скуп података, па се самим тим може очекивати да перформансе креираног стабла одлучивања могу бити за који проценат слабије приликом примене на непознатом тест скупу података. Управо из ових разлога, потребно је пронаћи простије стабло одлучивања које ће имати боље могућности примене над непознатим скупом података у односу на креирано сложеније стабло одлучивања. Како би се обавио процес одсецања извршено је израчунаване оптималног нивоа одсецања. За различите подскупове стабла одлучивања вршено је израчунавање *resubstitution* грешке и грешке унакрсне валидације. Поређење ових двеју вредности грешака за изачунати оптимални број нивоа у оквиру стабла одлучивања приказано је на слици 73. У случају оптималног броја нивоа, вредност *resubstitution* грешке је мало мања у односу на грешку унакрсне валидације.



Слика 73: Поређење вредности resubstitution *грешке и грешке унакрсне валидације након одсецања*

Вредност ове грешке, готово увек, расте са порастом стабла, међутим када се ради о односу грешака унакрсне валидације са смањењем стабла одлучивања, смањује се и однос грешака унакрсне валидације. Након одсецања стабла, најпростије правило селектовања стабла јесте одабир стабла са најмањом вредношћу грешке унакрсне валидације. Ово значи да се мора извршити израчунавање тачке одсецања. Тачка одсецања је једнака минималној вредности трошкова и додатно стандардна грешка која се може јавити. Оптимални број терминалних чворова у оквиру стабла одлучивања који води оптималном броју нивоа у оквиру креираног стабла одлучивања приказан је на слици 74.

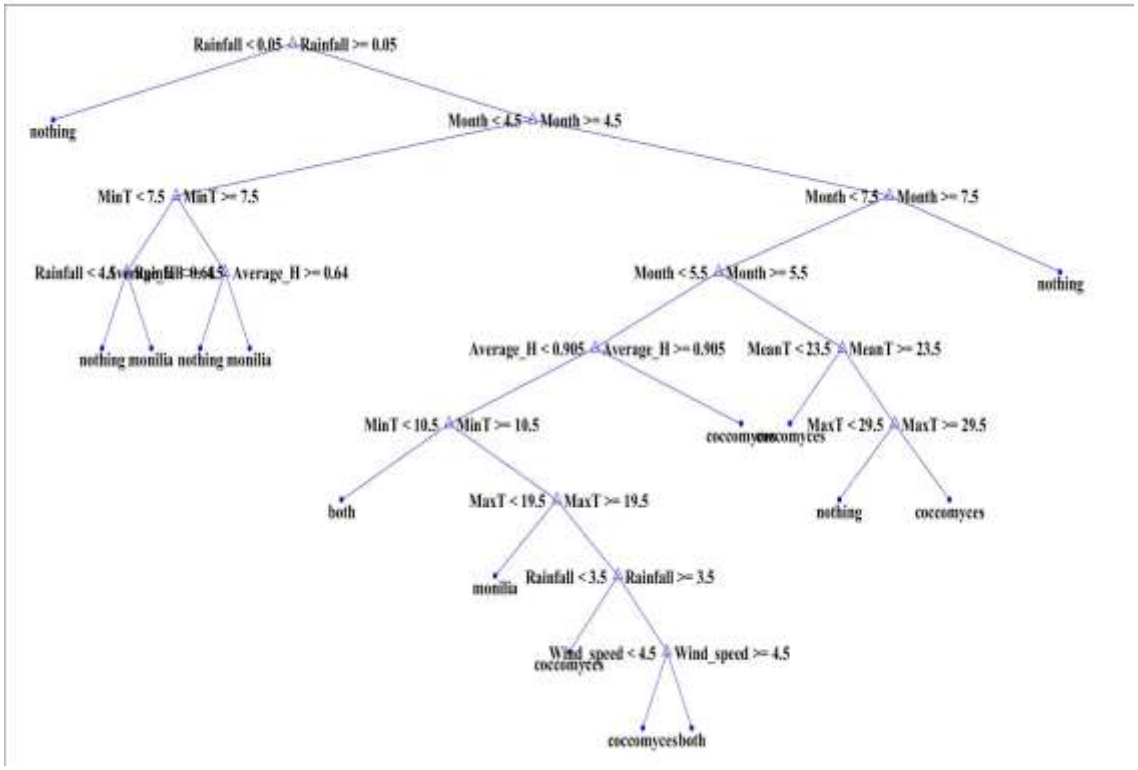


Слика 74: Репрезентација оптималног броја терминалних чворова

Оптимални број терминалних чворова одсеченог стабла одлучивања, добијен израчунавањем и представљен на слици 74, износи тринаест чворова. На основу овако добијеног броја терминалних чворова је извршена регулација броја нивоа у оквиру стабла одлучивања. Одсецањем стабла одлучивања, првобитно креирано стабло одлучивања које се састојало од десет нивоа, редуковано је на структуру од осам нивоа.

Стабло одлучивања, добијено одсецањем два нивоа у складу са добијеним оптималним бројем терминалних чворова, приказано је на слици 75. Овако креирано одсечено стабло одлучивања обезбеђује креирање одређеног броја предикционих правила. Свако од предикционих правила представља једну путању у стаблу одлучивања. Корени чвор стабла одлучивања представља атрибут који носи информацију о количини падавина. Вредност дневне количине падавина у оквиру кореног чвора стабла одлучивања указује на чињеницу да је за остварење услова за инфекцију гајених биљака посматраним болестима потребна одређена количина подавина. Уколико на дневном нивоу нема падавина, неће доћи ни до остварености услова за појаву инфекције, а самим тим нема потребе ни за хемијским третманима. Следећи параметар који је од значаја у процесу предикције јесте временски податак који представља информацију о томе у ком временском тренутку су прикупљени метеоролошких подаци. Овај параметар је редукован и односи се на месец коме припадају дати подаци. Уколико се посматра креирано стабло одлучавања, уочљиво је да временски параметар врши поделу стабла одлучивања на два подстабла. Свако од овако креираних подстабала одлучивања представља скуп правила одлучивања на којима се базира предикција времена хемијских третмана за две посматране биљне болести. Даље гранање креираног стабла одлучивања се заснива на вредностима осталих посматраних метеоролошких параметара. На овакав начин су вредности зависног атрибута директно условљене вредностима неких од независних атрибута. Тачније, у процесу формирања вредности зависног атрибута, са једне стране могу учествовати сви независни атрибути, док са друге стране вредност зависног атрибута може бити формирана само на основу вредности појединих независних атрибута. Ово значи да се, почевши од кореног чвора стабла, са сваким гранањем врши провера испуњености услова гранања, па се самим тим у зависности од испуњености дефинисаног услова може одредити вредност независног параметра без потребе за даљим гранањем. На пример, након гранања на лево подстабло, у зависности од вредности месеца, врши се даље гранање у коме фигурира вредност минималне дневне температуре, након чега се у зависности од обављеног гранања врши гранање у односу на вредност количине падавина или у односу на вредност просечне влажности ваздуха.

Тако, у процес предикције вредности зависног атрибута нису укључене вредности које представљају остале метеоролошке параметре (максимална температура ваздуха, средња дневна температура ваздуха и брзина ветра). Применом истог принципа, уколико је вредност количине падавина мања од 0,05 mm, нема остварености услова за појаву инфекције, па нема потребе за провером осталих метеоролошких параметара, већ се само на основу вредности овог параметра може одредити предиктивна вредност зависног атрибута.



Слика 75: Репрезентација стабла одлучивања након одсецања

Повећање степена тачности предикције засноване на предикционом моделу, базираном на стаблу одлучивања креираном применом J48 алгоритма, израчунато је поређењем тачности овог предикционог модела пре процеса одсецања и након процеса одсецања. Процес процене тачности предикционог модела је поновљен коришћењем истоветног тест скупа података након завршеног процеса одсецања креираног стабла одлучивања. Тачност предикције, базиране на новокреираном стаблу одлучивања, износи 93,71% што представља успешнију предикцију за 2,2% у односу на тачност првобитно креираног стабла одлучивања од 91,51%.

Како се овако креирани предикциони модел одликује највећом предикционом тачношћу у односу на све остале креиране предикционе моделе, може се сматрати да предикциони модел заснован на J48 стаблу одлучивања најбоље осликава зависности међу подацима на којима се базира креирање предикционих правила, а самим тим и процес предикције. Тачност осталих креираних предикционих модела није занемарљива, што значи да и њихова примена у процесу предикције вредности зависног атрибута може довести до задовољавајућих резултата.

С обзиром на чињеницу да се предикциони модел, креиран коришћењем J48 алгоритма, одликује највећим степеном тачности, извршено је детаљније тестирање тачности овог модела. Креирано стабло одлучивања овог модела, настало након процеса одсецања, као што се може видети на слици 75, садржи петнаест потенцијалних путања. Свака од креираних петнаест путања у оквиру стабла одлучивања одговара једном правилу одлучивања које недвосмислено води до једне од четири могуће вредности зависног атрибута. Процес предикције вредности зависног атрибута се заснива на праћењу генерисаних правила одлучивања. На овакав начин, поређењем вредности независних атрибута у оквиру инстанци података (инстанце података за које је потребно одредити вредности зависног арибута) са вредностима у оквиру чворова стабла одлучивања у којима долази до гранања стабла, врши се одабир путање до вредности зависног атрибута коју представља терминални чвор стабла одлучивања.

Креирано стабло одлучивања и предности овако организованог процеса предикције су искоришћени у поступку потврде тачности креираног предикционог модела. Потврда тачности креираног предикционог модела, поред тестирања коришћењем тест скупа података са познатим исходом у оквиру истраживања, извршена је и поређењем креираних предикционих правила са познатим чињеницама доступним у референтној литератури. Одабрана референтна литература припада области фитопатологије и садржи битне чињенице о појави и развоју посматраних двеју болести. Процес развоја посматраних болести је у оквиру коришћене литературе дефинисан кроз праћење параметара потребних за појаву посматраних болести, њихов развој и потребних метеоролошких услова за остварење инфекције гајених биљка.

Како је циљ креираног предикционог модела његово коришћење у процесу предикције времена хемијских третмана која је у директној вези са остварношћу услова за појаву инфекције, параметри од значаја у оквиру референтне литературе јесу параметри који се односе на услове потребне за инфекцију гајених биљака. Потребни услови за појаву датих болести и остварење инфекције у оквиру референтне литературе су потврђени, како лабораторијским испитивањима тако и експериментима спроведеним на производним површинама. Потребни услови за инфекцију гајених биљака јесу одговарајући метеоролошки услови. Вредности посматраних метеоролошких параметара потребних за остварење инфекције у оквиру референтне литературе претежно су дати у одређеним опсезима. Сваки од опсега је дефинисан у склопу неког од спроведених истраживања, па се може сматрати адекватним за поређење са вредностима метеоролошких параметара у оквиру генерисаног стабла одлучивања. Дефинисање вредносних опсега за посматране метеоролошке параметре указује на чињеницу да још увек у области фитопатологије нису документовани конкретни услови потребни за остварење инфекције посматраним болестима.

Како би се извршило поређење генерисаних правила одлучивања са подацима из референтне литературе креирана је табела 19, у оквиру које су сумирани параметри дефинисани у литератури и вредности ових параметара дефинисани у оквиру креираних предикционих правила. Тако на пример, резултати доступни у оквиру референтне литературе, добијени испитивањима у лабораторијским условима, показују да је за појаву посматраних болести и њихов развој на *PDA* медијуму потребна температура у опсегу од 5°C до 30°C. На основу експеримента се може утврдити да до периода инкубације може доћи уколико је средња дневна температура између 17.5°C и 23°C. Такође, период инкубације траје у просеку од 2 до 4 дана. Ово значи да од момента инфекције до појаве првих видљивих симптома прође око 4 дана. Оптимална влажност ваздуха у лабораторијским испитивањима је износила 100%, док је у испитивањима спроведеним у природним условима износила изнад 85%. Минимална дневна количина падавина потребна за појаву инфекције износила је изнад 1 mm воденог талога.

Табела 19: Поређење услова за појаву болести из референтне литературе и резултата добијених предикцијом

Референтна литература	**Мин. температура ваздуха [ºC]**	**Мах. температура ваздуха [ºC]**	**Средња дневна температура ваздуха [ºC]**	**Просечна влажност ваздуха [%]**	**Укупна количина падавина [mm]**	**Брзина ветра [km/h]**	**Редни број месеца у години**	**Обољење *monilia***	**Обољење *Coccomyces***
Референца [1]	7	26	16-23	85	>1	/	4-6	+	+
Референца [2]	5	27	15-20	/	>3	/	5	+	+
Референца [3]	7-11	16	20	>85	>1	10	>4	+	+
Референца [4]	5	30	17.5-23	>85	>1	>5	5	+	+
Предикционо правило	>=10,5	>=19,5	/	<90	>=3,5	>=4,5	<5,5	+	+
Предикционо правило	>7,5	/	/	>=64	>=0,05	/	<4,5	+	-
Предикционо правило	>=10,5	>=19,5	/	<90	<3,5	/	<5,5	-	+
Предикционо правило	<7,5	/	/	/	<4,5	/	<4,5	-	-
Предикционо правило	/	/	/	/	>=0,05	/	>=7,5	-	-
Предикционо правило	>=10,5	<19,5	/	<90	>=0,05	/	<5,5	+	-
Предикционо правило	/	<29,5	>=23,5	/	>=0,05	/	>=5,5	-	-
Предикционо правило	>=10,5	>=19,5	/	<90	>=3,5	<4,5	<5,5	-	+
. . .	. . .	. . .	. . .	. . .	. . .	. . .		. . .	. . .

Правила одлучивања наведена у оквиру табеле 19 представљају део од укупног броја креираних правила одлучивања. Поређењем наведених правила одлучивања са информацијама о потребним метеоролошким условима за остварење инфекције добијена су поклапања за сваки од посматраних параметра. На пример, према референтној литератури минимална температура за остварење инфекције треба да буде у опсегу од 7°C до 11°C, док у оквиру наведених правила одлучивања потребна минимална температура ваздуха износи 10,5°C и више од 10,5°C. Истовремено, правилима одлучивања је дефинисано да, уколико је минимална температура ваздуха мања од 7,5°C, нема остварености услова за појаву инфекције, што потврђују и подаци доступни у оквиру коришћене референтне литературе. Максимална температура ваздуха у референтној литератури је у опсегу од 16°C до 30°C, док је у оквиру правила одлучивања дефинисано да до остварености услова за појаву инфекције са становишта максималне дневне температуре ваздуха може доћи уколико је она једнака или изнад 19,5°C. Средња дневна температура ваздуха у оквиру креираних предикционих правила најчешће не фигурира као параметар. Међутим, уколико се за иста предикциона правила посматра разлика између минималне и максималне дневне температуре ваздуха, може се установи да је и средња дневна температура ваздуха у опсезима дефинисаним у референтној литератури. Просечна влажност ваздуха у оквиру референтне литературе, као што је раније наведено, износи око 85% или изнад 85%. У оквиру креираних предикционих правила, просечна влажност ваздуха је дефинисана у опсегу изнад 64% а испод 90%, што представља поклапање са дефинисаним вредностима у оквиру референтне литературе. Минимална количина падавина је дефинисана у референтној литератури, као што је раније наведено, не више од 1 mm воденог талога. У оквиру креираних правила одлучивања вредност укупне минималне количине падавина представља корени чвор и износи 0,05 mm. Иако је ова вредност мања од референтних вредности, у већини путања стабла одлучивања врши се измена дате вредности на више од 3,5 mm воденог талога, што одговара опсегу дефинисаном у референтној литератури. Брзина ветра у оквиру референтне литературе је дефинисана на више од 5 km/h, што се поклапа са дефинисаном брзином ветра у оквиру креираних правила одлучивања која износи више или једнако од 4,5 km/h.

Временски параметар, дефинисан ознаком месеца коме припада свака од инстанци, указује да је могуће остваривање услова за инфекцију почевши од априла месеца закључно са јуном месецом. Такође, овај параметар показује да након јула месеца није могућа појава болести, без обзира на остварености свих осталих метеоролошких услова, о чему је и било речи у претходним поглављима.

Поређењем правила одлучивања у оквиру креираних предикционих модела са резултатима из референтне литературе, извршена је не само потврда успешности овако организоване предикције, већ и потврда успешности коришћења софтверских решења у процесу предикције времена хемијских третмана. Обједињавањем свих дефинисаних и изведених провера тачности креирнаних предикционих модела, долази се до закључка да је креирање софтверског система намењеног крајњим корисницима од изузетног значаја у будућој примени за поребе придикције времена хемијских третмана у оквиру пољопривредне производње. Поред компоненте овог система задужене за процес предикције, важно место заузима и комуникација система са метеоролошким станицама и клијентима система, заснована на коришћењу бежичних комуникационих система. Процес прикупљања података помоћу система метеоролошких станица, повезаних коришћењем бежичних комуникационих система, од кључног је значаја за процес предикције. Оваквим системом се обезбеђује ажурираност података и њихово прикупљање без одласка на саму локацију метеоролшке станице, што у великој мери доприноси тачности предикције и брзој и ефикасној употреби креираног система.

11. Закључак

Доступност информационо комуникационих технологија, предности њихове употребе, као и могућности примене у различитим областима свакодневног живота и рада људи су допринеле унапређењу великог броја животних активности. Евидентан је пораст хардверских и софтверских решења намењених примени у пољопривредној производњи. Управо из ових разлога се сматра да је примена информационо комуникационих технологија у пољопривредној производњи једна од области са најбржим развојем. Велики проценат пољопривредних активности је данас готово незамислив без примене софверских решења. Информационо комуникационе технологије, поред индивидуалне примене, нашле су своје место и у комбинацији са великим компанијама произвођачима машина и опреме које се користе у пољопривредној производњи. Примена оваквих решења се одликује прецизношћу и квалитетом обављања пољопривредних активности, као и независношћу пољопривредних процеса од људског фактора. Такође, примена информационо комуникационих решења у пољопривредној производњи, посматрано из угла пољопривредних произвођача, значајно олакшава сами процес производње.

Велики број информационо комуникационих решења, применљивих у пољопривредној производњи, има за задатак реализацију система за доношење одлука. Системи за доношење одлука се у највећем броју случајева базирају на примени data mining техника. Применом data mining техника се обавља обрада великих скупова података, у оквиру којих се налазе потребне информације на основу којих је могуће донети одлуку о времену и поступку обављања неке пољопривредне активности. Обрада поменутих скупова података, дефинисање веза између података и доношење одлука базираних на везама међу подацима, незамисливо је обавити без примене data mining техника.

Једна од области примене data mining техника, са циљем унапређења процеса пољопривредне производње, производње здравије хране и заштите животне средине, јесте и одређивање времена хемијских третмана. Хемијски третмани, као основни механизам заштите гајених биљка од великог броја патогена/штеточина, неизоставан су део пољопривредне производње.

Процес одређивања времена хемијских третмана је заснован на посматрању великог броја фактора. Основни скуп фактора потребних за предикцију времена хемијских третмана представљају метеоролошки и просторно-временски параметри. Метеоролошки и просторно-временски параметри су лако доступни пољопривредним произвођачима, па се употребом адекватног софтверског решења може успешно извршити предикција времена хемијских третмана.

Основни циљеви дефинисани овом докторском дисертацијом су се односили на креирање система којим би било омогућено активно прикупљање метеоролошких и просторно-временских података као и предикција времена хемијских третмана заснована на прикупљеним подацима. С обзиром на чињеницу да се метеоролошки услови могу разликовати у зависности од локалитета, креиран је модел метеоролошке станице намењен употреби у пољопривреди. Креирани модел се састоји од метеоролошких и просторно-временских мерних компоненти одговарајућих перформанси намењених за рад у спољашњем окружењу. Одабир одговарајућих компоненти је извршен поређењем компоненти доступних на тржишту, на основу унапред дефинисаних критеријума. Мобилност, моделом предложене метеоролошке станице, обезбеђена је пројектовањем напајања помоћу фотонапонског панела и слањем података коришћењем бежичних телекомуникационих мрежа. Одабране компоненете обезбеђују креирање метеоролошке станице економски доступније индивидуалном пољопривредном произвођачу, у односу на комерцијалне метеоролошке станице доступне на тржишту.

Вредности метеоролошких и просторно-временских параметара, прикупљене на дневном нивоу, нуде значајну количину информација, на основу којих се може спровести анализа остварености услова за појаву болести, а самим тим и времена хемијских третмана. Други циљ ове дисертације је било крерирање софтверског решења чији је задатак обрада метеоролошких и просторно-временских скупова података, креирање и обука предикционих модела, заснованих на датим скуповима података, као и коришћење креираних предикционих модела у процесу предикције времена хемијских третмана.

Применом имплементираних функционалности у оквиру креираног софтверског решења извршено је тестирање успешности креираних предикционих модела применом већег броја евалуационих метода. Процена успешности предикционих модела је извршена и поређењем дефинисаних правила одлучивања са чињеницама доступним у референтној литератури. Израчуната тачност креираних предикционих модела се разликовала од модела до модела. Без обзира на разлику о оствареној тачности, већина креираних предикционих модела је остварила висок степен тачности. Креирано софтверско решење, као крајњи продукт дисертације, потпуно је функционално и намењено употреби од стране крајњих корисника.

Кроз креирање модела метеоролошке станице, намењене употреби у процесу пољопривредне производње, као и креирањем софтверског решења, намењеног обради прикупљених података и предикцији времена хемијских третмана, осварени су задати полазни циљеви ове дисертације. Такође, проценом тачности креираних предикционих модела и поређењем са познатим чињеницама из референтне литературе, потврђена је и полазна хипотеза да се применом data mining-а и бежичних комуникационих система може успешно извршити предикција времена хемијских третмана у пољопривредној производњи.

Креирано софтверско решење се користити као *stand alone* верзија на рачунарима корисника. Како би се обезбедила већа доступност креираног софтверског решења, а самим тим и употреба од стране већег броја корисника, једна од идеја за наставак истраживања јесте доступност креираног софтверског решења у форми *web* портала. Доступност софтверског решења у форми *web* портала представља својеврсну нову имплементацију и примену додатних технологија чиме се планира даље усавршавање и унапређење система.

12. Литература

[1] M. Ivanović, *Mikoze biljaka*. Beograd: Nauka, 1992.

[2] S. Stojanović, *Poljoprivredna fitopatologija*. Kragujevac: Srpsko biološko društvo, Stevan Jakovljević, 2004.

[3] R. Byrde and J. Willets, *The Brown Rot Fungi of Fruit: Their Biology and Control*. Oxford, England: Pergamon, 1977.

[4] S. Perić, "Efficiency of fungicides used with low-risk preparations and importance of mechanical measures in eradication control of pathogenic monilinia laxa," University of Pristina, 2007.

[5] L. Rokach and O. Maimon, "Decision Trees," in *Data Mining and Knowledge Discovery Handbook*, Second., O. Maimon and L. Rokach, Eds. London: Springer, 2010, pp. 165–192.

[6] M. Kantardzic, *Data Mining: Concepts, Models, Methods, and Algorithms: Second Edition*. A John Wiley & Sons, Inc., Hoboken, New Jersey., 2011.

[7] S. Shekhar *et al.*, "Spatiotemporal Data Mining: A Computational Perspective," *ISPRS Int. J. Geo-Information*, vol. 4, no. 4, pp. 2306–2338, 2015.

[8] O. Janković, "Data Mining : Implikacije wrapper pristupa selekcije atributa na performanse klasifikacionog modela," vol. 15, no. March, pp. 660–664, 2016.

[9] J. Novaković, *Solving Machine Learning Classification Problems*, vol. 4. Faculty of Technical Sciences Cacak University of Kragujevac, 2013.

[10] N. Ajzenhamer, A. Bukurov, and V. Stanković, *Istraživanje podataka*. Matematički fakultet, Univerzitet u Beogradu, Beograd, 2017.

[11] V. R. Minić, P. I. Lugonja, and V. S. Crnojević, "A complete system for remote collection of meteorological data," in *Proceedings of 19th Telecommunications Forum, TELFOR*, 2011, pp. 20–22.

[12] K. Krishnamurthi, S. Thapa, L. Kothari, and A. Prakash, "Arduino Based Weather Monitoring System," *Int. J. Eng. Res. Gen. Sci.*, vol. 3, no. 2, pp. 452–458, 2015.

[13] J. Song, "Greenhouse Monitoring and Control System Based on Zigbee Wireless Senor Network," in *Proceedings of International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering*, 2015, pp. 1–4.

[14] A. Fourati, M.A. Chebbi, W. Kamoun, "Development of a web-based weather station for irrigation scheduling," in *Third IEEE International Colloquium in Information Science and Technology (CIST)*, 2014, pp. 37–42.

[15] S. Tenzin, S. Siyang, T. Pobkrut, and T. Kerdcharoen, "Low Cost Weather Station for Climate-Smart Agriculture," in *Proceedings of 9th International*

Conference on Knowledge and Smart Technology (KST), 2017, pp. 172–177.

[16] N. Watthanawisuth, A. Tuantranont, and T. Kerdcharoen, "Microclimate real-time monitoring based on ZigBee sensor network," in *Proceedings of IEEE Sensors*, 2009, pp. 1814–1818.

[17] V. Vishwarupe, M. Bedekar, and S. Zahoor, "Zone specific weather monitoring system using crowdsourcing and telecom infrastructure," in *Proceedings of IEEE International Conference on Information Processing, ICIP 2015*, 2015, pp. 823–827.

[18] M. A. Joshi, M. R. Jathar, and S. C. Mehrotra, "Distributed System for Weather Data Collection through TINI Microcontroller," *Int. J. Environ. Sci. Dev.*, vol. 2, no. 1, pp. 70–72, 2011.

[19] C.-L. Tseng *et al.*, "Feasibility study on application of GSM–SMS technology to field data acquisition," *Comput. Electron. Agric.*, vol. 53, no. 1, pp. 45–59, 2006.

[20] J.-A. Jiang *et al.*, "A GSM-based remote wireless automatic monitoring system for field information: A case study for ecological monitoring of the oriental fruit fly, Bactrocera dorsalis (Hendel)," *Comput. Electron. Agric.*, vol. 62, no. 2, pp. 243–259, 2008.

[21] C.-L. Chuang and J.-A. Jiang, "ICT-based Remote Agro-Ecological Monitoring System A Case Study inTaiwan," *J. Commun. Navig. Sens. Serv.*, vol. 1, no. 1, pp. 67–92, 2014.

[22] W. Chebbi, M. Benjemaa, A. Kamoun, M. Jabloun, and A. Sahli, "Development of a WSN integrated weather station node for an irrigation alert program under Tunisian conditions," in *Proceedings of International Multi-Conference on Systems, Signals and Devices, SSD'11*, 2011, no. figure 1.

[23] P. Susmitha and G. Sowmyabala, "Design and Implementation of Weather Monitoring and Controlling System," *Int. J. Comput. Appl.*, vol. 97, no. 3, pp. 975–8887, 2014.

[24] T. Parashar, S. Gahlot, A. Godbole, and Y. Thakare, "Weather Monitoring System Using Wi-Fi," *Int. J. Sci. Res.*, vol. 5, no. 11, pp. 891–893, 2016.

[25] L. J. Olatomiwa and U. S. Adikwu, "Design and Construction of a Low Cost Digital Weather Station," *AU J. Technol.*, vol. 16, no. 2, pp. 125–132, 2012.

[26] X. Guo and Y. Song, "Design of automatic weather station based on GSM module," in *Proccedings of International Conference on Computer, Mechatronics, Control and Electronic Engineering, CMCE*, 2010, vol. 5, pp. 80–82.

[27] H. S. Bagiorgas, M. N. Assimakopoulos, A. Patentalaki, N. Konofaos, D. P. Matthopoulos, and G. Mihalakakou, "The design, installation and operation of a fully computerized, automatic weather station for high quality meteorological measurements," *Fresenius Environ. Bull.*, vol. 16, no. 8, pp. 948–962, 2007.

[28] P. A. Kulkarni and V. V Yerigeri, “An economical weather monitoring system based on GSM using solar and wind energy,” *Int. J. Adv. Technol. Innov. Res. Vol.*, vol. 7, no. 02, pp. 0263–0268, 2015.

[29] S. H. Parvez *et al.*, “A Novel Design and Implementation of Electronic Weather Station and Weather Data Transmission System Using GSM Network,” *WSEAS Trans. CIRCUITS Syst.*, vol. 15, pp. 21–34, 2016.

[30] F. Paulose, A. Mathew, and G. George, “GPS / GSM Based Embedded System for Atmospheric Boundary Layer Profiling and Weather Monitoring,” *Int. J. Sci. Res.*, vol. 3, no. 9, pp. 2319–7064, 2012.

[31] S. Chakraborty, R. Ghosh, M. Ghosh, C. D. Fernandes, M. J. Charchar, and S. Kelemu, “Weather-based prediction of anthracnose severity using artificial neural network models,” *Plant Pathol.*, vol. 53, no. 4, pp. 375–386, 2004.

[32] M. S. P. Babu and B. S. Rao, “Leaves Recognition Using Back Propagation Neural Network-Advice for Pest & Disease Control on Crops . Leaves Recognition Using Back Propagation Neural Network-Advice for Pest & Disease Control on Crops . 3 . Requirement Specification 4 . Tools Techniques,” 2007.

[33] K. S. Kim, T. C. Wang, and X. B. Yang, “Simulation of apparent infection rate to predict severity of soybean rust using a fuzzy logic system.,” *Phytopathology*, vol. 95, no. 10, pp. 1122–1131, 2005.

[34] K. S. Kim, M. L. Gleason, and S. E. Taylor, “Forecasting Site-Specific Leaf Wetness Duration for Input to Disease-Warning Systems,” *Plant Dis.*, vol. 90, no. 5, pp. 650–656, 2006.

[35] G. Liu, H. Shen, X. Yang, and Y. Ge, “Research on prediction about fruit tree diseases and insect pests based on neural network,” in *Artificial Intelligence Applications and Innovations. AIAI 2005. IFIP — The International Federation for Information Processing*, Vol 187., W. B. Li D., Ed. Boston, MA: Springer, 2005, pp. 731–740.

[36] L. K. Mehra, C. Cowger, K. Gross, and P. S. Ojiambo, “Predicting Pre-planting Risk of Stagonospora nodorum blotch in Winter Wheat Using Machine Learning Models,” *Front. Plant Sci.*, vol. 7, no. March, pp. 1–14, 2016.

[37] R. Kaundal, A. S. Kapoor, and G. P. S. Raghava, “Machine learning techniques in disease forecasting: a case study on rice blast prediction.,” *BMC Bioinformatics*, vol. 7, no. 1, p. 485, 2006.

[38] M. G. Hill, P. G. Connolly, P. Reutemann, and D. Fletcher, “The use of data mining to assist crop protection decisions on kiwifruit in New Zealand,” *Comput. Electron. Agric.*, vol. 108, pp. 250–257, 2014.

[39] A. K. Tripathy *et al.*, “Data mining and wireless sensor network for agriculture pest/disease predictions,” in *Proceedings of the 2011 World Congress on Information and Communication Technologies, WICT 2011*, 2011, pp. 1229–1234.

[40] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques,” *Informatica*, vol. 31, pp. 249–268, 2007.

[41] M. Radovanović, “High-Dimensional Data Representations and Metrics for Machine Learning and Data Mining,” University of Novi Sad, Faculty of Science, 2010.

[42] H. Almuallim and T. G. Dietterich, “Efficient algorithms for identifying relevant features,” in *Proccedings of 9th Canadian Conference on Artificial Intelligence*, 1992, pp. 38–45.

[43] C. Cardie, “Using Decision Trees to Improve Case-Based Learning.,” in *Proceedings of 10th International Conference on Machine Learning*, 1993, pp. 25–32.

[44] M. Singh and G. M. Provan, “Efficient learning of selective Bayesian network classifiers,” in *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, 1996, pp. 453–461.

[45] P. Langley and S. Sage, “Scaling to domains with irrelevant features,” in *Computational Learning Theory and Natural Learning Systems*, vol. IV: Making, R. Greiner, T. Petsche, and S. Hanson, Eds. London: The MIT Press, 1997, pp. 51–63.

[46] J. R. Quinlan, “Induction of Decision Trees,” *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[47] R. Quinlan, *Book Review : C4 . 5 : Programs for Machine Learning*, vol. 240. Boston, 1994.

[48] R. Kohavi, “Wrappers for Performance Enhancement and Oblivious Decision Graphs,” Stanford University, 1995.

[49] M. J. Pazzani, “Searching for dependencies in Bayesian classifiers,” in *Learning from Data. Lecture Notes in Statistics*, D. Fisher and H. Lenz, Eds. New Your: Springer, 1996, pp. 239–248.

[50] J. Shlens, “A Tutorial on Principal Component Analysis,” *arXiv Prepr. arXiv1404.1100.*, 2014.

[51] M. Ilic, S. Ilic, S. Jovic, and S. Panic, “Early cherry fruit pathogen disease detection based on data mining prediction,” *Comput. Electron. Agric.*, vol. 150, no. May, pp. 418–425, 2018.

[52] E. Frank and I. H. Witten, “Making better use of global discretization,” in *Proceedings of the 16th International Conference on Machine Learning*, 1999, pp. 115–123.

[53] R. Kerber, “Chimerge: Discretization of numeric attributes,” in *Proceedings of the tenth national conference on Artificial intelligence*, 1992, pp. 123–128.

[54] S. D. Bay, “Multivariate discretization of continuous variables for set mining,” in

Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, 2000, pp. 315–319.

[55] Y. Yang and G. I. Webb, "Discretization for naive-Bayes learning: Managing discretization bias and variance," *Mach. Learn.*, vol. 74, no. 1, pp. 39–74, 2009.

[56] C. Hsu, "01 - Implications of the Dirichlet Assumption for Discretization of Continuous Variables in Naive Bayesian Classifiers.pdf," *Mach. Learn.*, vol. 53, no. 3, pp. 235–263, 2003.

[57] H. Ishibuchi, T. Yamamoto, and T. Nakashima, "Fuzzy data mining: effect of fuzzy discretization," in *Proceedings 2001 IEEE International Conference on Data Mining*, 2001, no. 2, pp. 241–248.

[58] P. D. Allison, *Missing Data*. London: Sage Publications, Inc, 2002.

[59] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*. , Hoboken, New Jersey.: John Wiley & Sons, Inc., Hoboken, New Jersey., 2002.

[60] P. Clark and T. Niblett, "The CN2 Induction Algorithm," *Mach. Learn.*, vol. 3, no. 4, pp. 261–283, 1989.

[61] J. W. Grzymala-Busse and M. Hu, "A Comparison of Several Approaches to Missing Attribute Values in Data Mining," in *Rough sets and current trends in computing*, 2001, vol. 2005, no. Chapter 46, pp. 378–385.

[62] V. Blahut, "Outlier Detection and Explanation," Masaryk University, Faculty of Informatics, 2015.

[63] K. Kaur and A. Garg, "Comparative Study of Outlier Detection Algorithms," *Int. J. Comput. Appl.*, vol. 147, no. 9, pp. 21–26, 2016.

[64] J. I. Maletic and A. Marcus, "Data Cleansing: A Prelude to Knowledge Discovery," in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. London: Springer, 2010, pp. 19–32.

[65] K. Bhaduri, B. L. Matthews, and C. Giannella, "Algorithms for speeding up distance-based outlier detection," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, 2011, pp. 859–867.

[66] K. Singh and S. Upadhyaya, "Outlier Detection: Applications And Techniques.," *Int. J. Comput. Sci.*, vol. 9, no. 1, pp. 307–323, 2012.

[67] H. Richard, *Coding and Information Theory*. New Jersey: Prentice-Hall, 1980.

[68] W. Fan, M. Miller, S. Stolfo, W. Lee, and P. Chan, "Using artificial anomalies to detect unknown and known network intrusions," in *Proceedings IEEE International Conference on Data Mining, IEEE Computer Society*, 2004, vol. 6, no. 5, pp. 507–527.

[69] Y. Shi and L. Zhang, "COID: A cluster-outlier iterative detection approach to

multi-dimensional data analysis," *Knowl. Inf. Syst.*, vol. 28, no. 3, pp. 709–733, 2011.

[70] C.-T. Lu, D. Chen, and Y. Kou, "Algorithms for spatial outlier detection," in *Proceedings of Third IEEE International Conference on Data Mining*, 2003, pp. 597–600.

[71] A. M. Said, D. D. Dominic, and B. B. Samir, "Frequent pattern-based outlier detection measurements: A survey," in *Proceedings of 2011 International Conference on Research and Innovation in Information Systems, ICRIIS'11*, 2011, pp. 1–6.

[72] Z. He, X. Xu, J. Z. Huang, and S. Deng, "FP-Outlier : Frequent Pattern Based Outlier Detection Related Work and Research Motivation," *Comput. Sci. Inf. Syst.*, vol. 1, no. 2, pp. 103–118, 2005.

[73] P. Chandore and P. Chatur, "Outlier Detection Techniques over Streaming Data in Data Mining : A Research Perspective," *Int. J. Recent Technol. Eng.*, vol. 2, no. 1, pp. 157–162, 2013.

[74] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *ACM SIGMOD International Conference on Management of Data*, 1993, vol. 22, no. 2, pp. 207–216.

[75] L.-J. Kao and Y.-P. Huang, "Association rules based algorithm for identifying outlier transactions in data stream," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2012, pp. 3209–3214.

[76] J. F. Kenney and E. S. Keeping, *Mathematics of Statistics*, 3rd ed., vol. Part 1. New Jersey, 1962.

[77] M. H. Kutner, C. Nachtsheim, J. Neter, and W. Li, *Applied Linear Statistical Models*. New York: McGraw-Hill/Irwin, 2005.

[78] N. H. Timm, *Applied Multivariate Analysis*. 2012.

[79] G. A. . Seber, *Multivariate Observations*. New York: John Wiley., 1984.

[80] R. . Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugen.*, vol. 7, pp. 179–188, 1936.

[81] M. Siotani, T. Hayakawa, and Y. Fujikoshi, *Modern multivariate statistical analysis: A graduate course and handbook.* New York: American Sciences Press Syracuse, 1985.

[82] J. R. Quinlan, "Simplifying decision trees," *Int. J. Man. Mach. Stud.*, vol. 27, no. 3, pp. 221–234, 1987.

[83] J. Quilan, "Decision trees and multi-valued attributes," in *Machine Intelligence*, no. 11, J. Richards, Ed. Oxford, England: Oxford Univ. Press, 1988, pp. 305–318.

[84] L. de Mantaras, “A distance-based attribute selection measure for decision tree induction,” *Mach. Learn.*, vol. 6, no. 1, pp. 81–92, 1991.

[85] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. New York: Chapman & Hall, 1984.

[86] C. Olaru and L. Wehenkel, “A complete fuzzy decision tree technique,” *Fuzzy Sets Syst.*, vol. 138, no. 2, pp. 221–254, 2003.

[87] J. Mingers, “An Empirical Comparison of Pruning Methods for Decision Tree Induction,” *Mach. Learn.*, vol. 4, pp. 227–243, 1989.

[88] F. Esposito, D. Malerba, and G. Semeraro, “A comparative analysis of methods for pruning decision trees,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 5, pp. 476–491, 1997.

[89] T. Hancock, T. Jiang, M. Li, and J. Tromp, “Lower Bounds on Learning Decision Lists and Trees,” *Inf. Comput.*, vol. 126, no. 2, pp. 114–122, 1996.

[90] L. Hyafil and R. L. Rivest, “Constructing Optimal Binary Trees in NP-Complete,” *Inf. Process. Lett.*, vol. 5, no. 1, pp. 15–17, 1976.

[91] H. Zantema and H. L. Bodlaender, “Finding small equivalent decision trees is hard,” *Int. J. Found. Comput. Sci.*, vol. 11, no. 2, pp. 343–354, 2000.

[92] A. Bhadgale, S. Natu, S. Deshpande, and A. Nilegaonkar, “Implementation of Improved ID3 Algorithm Based on Association Function,” *Int. J. Pure Appl. Math.*, vol. 114, no. 10, pp. 1–9, 2017.

[93] B. Pejčić, “Primena algoritma stabla odlučivanja u prepoznavanju ponašanja i zdravstvenih rizika kod starijih osoba,” Univerzitet u Beogradu, 2017.

[94] G. Kaur and A. Chhabra, “Improved J48 Classification Algorithm for the Prediction of Diabetes,” *Int. J. Comput. Appl.*, vol. 98, no. 22, pp. 13–17, 2014.

[95] G. Dimić, D. Prokin, K. Kuk, and M. Micalović, “Primena Decision Trees i Naive Bayes klasifikatora na skup podataka izdvojen iz Moodle kursa,” in *proceedings of Infoteh Jahorina*, 2012, no. 11, pp. 877–882.

[96] F. Provost and T. Fawcett, “Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions,” in *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, 1997, pp. 43–48.

[97] C. D. Manning, P. Ragahvan, and H. Schutze, *An Introduction to Information Retrieval*, Online edi., no. c. Cambridge, England: Cambridge University Press, 2009.

[98] N. Jardine and C. J. van Rijsbergen, “The use of hierarchic clustering in information retrieval,” *Inf. Storage Retr.*, vol. 7, no. 5, pp. 217–240, 1971.

[99] S. Mythili and Madhiya E, “An Analysis on Clustering Algorithms in Data

Mining," *Int. J. Comput. Sci. Mob. Comput.*, vol. 31, no. 1, pp. 334–340, 2014.

[100] C. Sammut and G. I. Webb, *Encyclopedia of Machine Learning and Data Mining*. Springer, 2016.

[101] L. Kaufman and P. Rousseeuw, *Finding Groups in Data An Introduction to Cluster Analysis*. Hoboken, New Jersey: John Wiley & Sons, Inc.

[102] Y. Zhao and G. Karypis, "Comparison of Agglomerative and Partitional Document Clustering Algorithms," Minneapolis, 2002.

[103] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, 1996, pp. 226–231.

[104] M. Verma, M. Srivastava, N. Chack, A. K. Diswar, and N. Gupta, "A Comparative Study of Various Clustering Algorithms in Data Mining," *Int. J. Eng. Res. Appl.*, vol. 2, no. 3, pp. 1379–1384, 2012.

[105] Y. Lv *et al.*, "An efficient and scalable density-based clustering algorithm for datasets with complex structures," *Neurocomputing*, vol. 171, pp. 9–22, 2016.

[106] V. Cariou, S. Verdun, E. Diaz, E. M. Qannari, and E. Vigneau, "Comparison of three hypothesis testing approaches for the selection of the appropriate number of clusters of variables," *Adv. Data Anal. Classif.*, vol. 3, no. 3, pp. 227–241, 2009.

[107] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *J. Intell. Inf. Syst.*, vol. 17, no. 2–3, pp. 107–145, 2001.

[108] T. Li, S. Ma, and M. Ogihara, "Entropy-based criterion in categorical clustering," in *Proccedings of Twenty-first international conference on Machine learning - ICML '04*, 2004, p. 68.

[109] N. Dragnić, "Konstrukcija i analiza klaster algoritma sa primenom u definisanju bihejvioralnih faktora rizika u populaciji odraslog stanovništva Srbije," Univerzitet u Novom Sadu, 2015.

[110] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun. Stat.*, vol. 3, no. 1, pp. 1–27, 1974.

[111] F. B. Baker and L. J. Hubert, "Measuring the Power of Hierarchical Cluster Analysis," *J. Am. Stat. Assoc.*, vol. 70, no. 349, pp. 31–38, 1975.

[112] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, no. C, pp. 53–65, 1987.

[113] L. Vendramin, R. J. G. B. Campello, and E. R. Hruschka, "Relative Clustering Validity Criteria : A Comparative Overview," *Stat. Anal. Data Min.*, vol. 3, pp. 209–235, 2010.

[114] M. Ilić, P. Spalević, and M. Veinović, "Predlog modela sistema za predikciju

pojave jutarnjih mrazeva," in *Zbornik radova 60. konferencije za elektroniku, telekomunikacije, računarstvo, automatiku i nuklearnu tehniku ETRAN*, 2016, p. RT3.6 1-6.

[115] D. Gislason, *Zigbee Wireless Networking*. 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA: Newnes - Elsevier, 2008.

[116] S. Farahani, *ZigBee Wireless Networks and Transceivers*. 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA: Newnes - Elsevier, 2008.

[117] O. Janković, "Korišćenje MPLAB IDE i PICDEM Z razvojnog okruženja," in *Zborniku radova Infoteh-Jahorina*, 2010, vol. 9, no. March, pp. 782–786.

[118] S. Draganić, "Analiza tehnologija kratkog dometa u funkciji IoT umrežavanja," Sveučilište u Zagrebu, 2016.

[119] D. International, "Zigbee networks: XBee/XBee-PRO S2C Zigbee RF module." Digi International Inc, 2018.

[120] M. Stojčev, "Celularne bežične mreže," in *Računarske mreže i prenos podataka*, Osnovni ud., Niš: Elektronski fakultet Niš, 2005, pp. 268–295.

[121] A. Tanenbaum, *Computer Networks*, Fifth edit. Boston, MA: Pearson Education, Inc, 2011.

[122] M. Stojčev, "TCP/IP." Elektronski fakultet Niš, Niš, pp. 58–88, 2007.

[123] M. Stojčev, "TCP/IP," in *Računarske mreže i prenos podataka*, Osnovni ud., Niš: Elektronski fakultet Niš, 2005, pp. 226–247.

[124] R. Technologies, "SIM 900 - TTL UART GSM/GPRS Modem." Rhydo Technologies (P) Ltd, Golden Plaza, Chitoor Road, Cochin – 682018, Kerala State, India, pp. 1–21, 2011.

[125] S. SIMCom, "TCPIP Application Note." Shanghai SIMCom Wireless Solutions Ltd, pp. 1–28, 2010.

[126] M. Meenakshi and G. Geetika, "Survey on Classification Methods using WEKA," *Int. J. Comput. Appl.*, vol. 86, no. 18, pp. 16–19, 2014.

[127] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. San Francisco: Morgan Kaufmann Publishers - Elsevier, 2005.

[128] B. Predic, M. Ilic, P. Spalevic, S. Trajkovic, S. Jovic, and A. Stanic, "Data mining based tool for early prediction of possible fruit pathogen infection," *Comput. Electron. Agric.*, vol. 154, no. September, pp. 314–319, 2018.

Биографија аутора

Милош Илић рођен је 11.10.1989. године у Прокупљу. Средњу школу – гимназија Прокупље, Природно математички смер завршио у Прокупљу 2008. године. Основне и дипломске-мастер академске студије на модулу Информационе технологије, студијског програма Рачунарство и информатика Електронског факултета у Нишу завршио је 2013. године и стекао стручни назив Мастер инжењер електротехнике и рачунарства – смер Информационе технологије. Исте 2013. године уписао је докторске студије на студијском програму Електротехничког и рачунарског инжењерства Факултета Техничких наука у Косовској Митровици. Од 07.03.2014. године Милош Илић ради као асистент на Високој пољопривредно прехрамбеној школи струковних студија у Прокупљу на групи предмета из области Информатика. Области интересовања Милоша Илића су: data mining, објектно оријентисано моделовање, вештачка интелигенција, обрада сигнала, сигурност података. Аутор је већег броја научних радова који су објављени у међународним и националним часописима, као и на међународним и националним скуповима, а чија је тематика директно везана за докторску дисертацију.

Прилог 1.

Изјава о ауторству

Потписани: Милош Илић

Број индекса: 3/2013

Изјављујем

да је докторска дисерација под називом

„Предикција времена хемијских третмана у пољопривредној производњи заснована на Data mining техници коришћењем бежичних комуникационих система“

- резултат сопственог истраживачког рада
- да предложена дисертација у целини ни у деловима није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио ауторска права и користио интелектуалну својину других лица.

Потпис докторанда

_____

Прилог 2.

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора: Милош Илић

Број индекса: 3/2013

Студијски програм: Електротехничко и рачунарско инжењерство

Наслов рада: Предикција времена хемијских третмана у пољопривредној производњи заснована на Data mining техници коришћењем бежичних комуникационих система

Ментор: проф. др Петар Спалевић

Потписани: Милош Илић

Изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао за објављивање на порталу **Дигиталног репозиторијума Универзитета у Приштини, са привременим седиштем у Косовској Митровици.**

Дозвољавам да се објаве моји лични подаци везани за добијање академског звања доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке у електронском каталогу и публикацијама Универзитета у Приштини, са привременим седиштем у Косовској Митровици.

Потпис докторанда

_____

У Косовској Митровици, _____

Прилог 3.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку да у Дигитали репозиторијум Универзитета у Приштини, са привременим седиштем у Косовској Митровици унесе моју докторску дисертацију под насловом:

„Предикција времена хемијских третмана у пољопривредној производњи заснована на Data mining техници коришћењем бежичних комуникационих система“

која је моје ауторско дело.

Дисертацију са свим прилозима предао сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитали репозиторијум Универзитета у Приштини са привременим седиштем у Косовској Митровици могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио.

1. Ауторство
2. Ауторство – некомерцијално
3. Ауторство – некомерцијално – без прераде
4. Ауторство – некомерцијално – делити под истим условима
5. Ауторство – без прераде
6. Ауторство – делити под истим условима

Потпис докторанда

_____

У Косовској Митровици, _____

1. Ауторство - Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.

2. Ауторство – некомерцијално. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.

3. Ауторство - некомерцијално – без прераде. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.

4. Ауторство - некомерцијално – делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.

5. Ауторство – без прераде. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.

6. Ауторство - делити под истим условима. Дозвољавате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.