Универзитет у Новом Саду
Факултет техничких наука у Новом Саду

Jelena Slivka

Adaptivni sistem za automatsku polu-nadgledanu klasifikaciju podataka

doktorska disertacija

Novi Sad 2014.

SADRŽAJ

Predgovor

Kategorizacija, odnosno klasifikacija, predstavlja organizaciju resursa u predefinisane, semantički smislene kategorije. Ona je od velikog značaja kako za organizaciju, tako i za analizu i razumevanje prikupljenih podataka. Danas se, zahvaljujući mnogim sistemima za automatsku kolekciju podataka, lako dolazi do velike količine podataka koji sadrže potencijalno vredne informacije i koje je neophodno analizirati. Usled izuzetno brzog rasta u količini i kompleksnosti prikupljenih podataka, automatizacija klasifikacije je postala presudna za funkcionisanje mnogih praktičnih sistema. Predmet istraživanja ove teze je razvoj adaptivnog sistema za automatsku klasifikaciju podataka.

Klasičan način formiranja automatskih sistema za klasifikaciju jeste *nadgledano obučavanje*, kod koga je sistemu potrebno pribaviti primere već klasifikovanih podataka, kako bi iz njih mogao da uči. Međutim, jedini način da se dođe do pouzdano klasifikovanih primera jeste ručna anotacija od strane eksperata domena, što ovaj proces čini skupim, a često i veoma dugotrajnim. Tehnike *polu-nadgledanog obučavanja* donekle olakšavaju ovaju problem smanjenjem količine anotiranih podataka neophodnih za obuku sistema. Ovo smanjenje se postiže tako što se u procesu obuke dodatno koriste i neanotirani podaci, koji, pod određenim uslovima, mogu pomoći da se stekne bolja slika o celokupnoj populaciji kojoj uočeni podaci pripadaju. Jedna od najpopularnije korišćenih tehnika polu-nadgledanog obučavanja jeste ko-trening algoritam. Međutim, primena ko-treninga je ograničena nametanjem određenih uslova koje podaci moraju zadovoljavati. Naime, sakupljeni podaci su u najvećem broju praktičnih domena opisani jedinstvenim skupom obeležja. Primena ko-treninga zahteva postojanje dva odvojena skupa obeležja koja opisuju podatke, koja bi u idealnom slučaju trebala biti uslovno nezavisna u odnosu na klasu [Blum 1998], odnosno, koja bi predstavljala dva nezavisna izvora informacija. Primer ovakve podele dat je u [Blum 1998] gde je za potrebe primene ko-trenina na problem automatske klasifikacije web stranica, za prvi skup obeležja koji opisuje web stranicu uzet skup reči koje se nalaze na samoj web stranici, a za drugi skup obeležja uzet skup reči iz linkova koji ukazuju na datu web stranicu. Iako se u nekim situacijama sreću dva nezavisna izvora informacija koja nameću ovakvu (prirodnu) podelu obeležja, ko-trening je zbog ovog uslova neprimenljiv u mnogim drugim situacijama gde bi bio od velike koristi.

Prethodni istraživači su generalno ovom problemu pristupali na dva načina. Prvi način jeste pokušaj pronalaženja veštačke podele obeležja koja bi rezultovala dobrim performansama ko-treninga. Međutim, ovo se pokazalo kao veoma težak zadatak jer su performanse ko-treninga veoma zavisne od korišćene podele, dok je karakteristike podele teško, ili čak nemoguće, verifikovati na malim skupovima podataka kakvim raspolažemo u ko-trening postavci. Drugi pristup ovom problemu jeste kombinovanje ko-treninga sa

tehnikama učenja sa grupom hipoteza. U tehnikama učenja sa grupom hipoteza se obučava više klasifikatora nakon čega se njihove predikcije kombinuju u cilju dobijanja finalne predikcije. Ovaj pristup je veoma efektivan ukoliko se za obučavanje klasifikatora koristi obučavajući algoritam koji je nestabilan, odnosno gde male promene u vrednosti odabranih parametara ili u korišćenom obučavajućem skupu mogu dovesti do velikih promena u izlaznoj hipotezi. Tako obučeni klasifikatori prave različite greške prilikom klasifikacije novih instanci, te mogu da uče jedan od drugog. Pristup kombinovanja ko-treninga sa tehnikama učenja sa grupom hipoteza se pokazao kao veoma uspešan, ali i značajno zahtevniji u pogledu veličine inicijalnog anotiranog skupa podataka.

U ovoj disertaciji predstavljena su dva adaptivna modela za automatsku klasifikaciju podataka, bazirana na ko-treningu. Cilj razvijenih modela je da omoguće primenu ko-treninga na skupove podataka kod kojih ne postoji definisana podela obeležja i pri tome povećaju performanse ko-trening algoritma. Kako bi se ovo postiglo, u predloženim rešenjima se ko-trening kombinuje sa tehnikama učenja sa grupom hipoteza. Za razliku od dosadašnjih tehnika baziranih na sličnom pristupu, modeli su razvijeni sa ciljem da zadrže važnu osobinu originalnog ko-treninga: da je za uspešnu primenu potreban samo izuzetno mali skup anotiranih primera.

Disertacija je organizovana u osam poglavlja. Prvo poglavlje, koje čine četiri odeljka, opisuje rešavani problem nedostataka anotiranih primera sa kojim se srećemo prilikom obučavanja automatskih sistema za klasifikaciju, opravdava potrebu za razvijenim rešenjem, izlaže ciljeve koje bi razvijeno rešenje trebalo da ispuni i u kratkim crtama izlaže opis razvijenog rešenja. U prvom odeljku ovog poglavlja opisani su modeli za nadgledano obučavanje, primenjeni u okviru ove disertacije. U drugom odeljku dat je opis genetskih algoritama, tehnike korišćene za optimizaciju parametara u jednom od razvijenih modela. U trećem odeljku ovog poglavlja dat je sažet pregled metoda polu-nadgledanog obučavanja u koje spada i razvijeno rešenje, sa posebnim akcentom na tehnologijama povezanim sa datim rešenjem. U četvrtom odeljku dat je pregled tehnika učenja sa grupom hipoteza koje su relevantne za razumevanje razvijenog rešenja.

Pregled postupaka kojima su prethodni autori pistupali rešavanju izloženog problema je dat u drugom poglavlju. Ovo poglavlje obrazlaže po čemu se izloženo rešenje razlikuje od postojećih rešenja i zbog čega postoji potreba za njim. Poglavlje je podeljeno u tri odeljka koja grubo odgovaraju postojećim pravcima istraživanja: kreiranje veštačke podele obeležja za primenu ko-treninga, kombinovanje ko-treninga sa tehnikama učenja sa grupom hipoteza i ostali pristupi koji se ne mogu svrstati u ove dve kategorije, ali su razvijeni sa istim ciljem primene ko-treninga na skupove podataka bez prirodne podele obeležja.

Formalan opis metodologije modela predstavljenih u ovoj disertaciji je dat u trećem poglavlju koje se sastoji od četiri odeljka. Prvi odeljak trećeg poglavlja opisuje kako se u modelima predloženim u ovoj disertaciji kreira grupa nezavisno obučenih klasifikatora (u cilju za primene tehnika učenja sa grupom hipoteza). Modeli predloženi u ovoj disertaciji, Algoritam statistike slučajnih podela i Integracija višestrukih ko-treninranih klasifikatora, opisani su u drugom, odnosno trećem odeljku trećeg poglavlja, respektivno. Četvrti odeljak trećeg poglavlja opisuje motivaciju koja stoji iza definisane metodologije predloženih rešenja.

Evaluacija razvijenih modela je prikazana u četvrtom poglavlju. Ovo poglavlje se sastoji od pet odeljaka. U prvom odeljku prikazan je pregled skupova podataka korišćenih u evaluaciji datih rešenja. Drugi odeljak izlistava i obrazlaže vrednosti parametara korišćene prilikom evaluacije sistema. Treći odeljak opisuje alternativne modele sa kojima se data rešenja porede u evaluaciji. Četvrti odeljak prikazuje rezultate evaluacije i njihovu diskusiju. Konačno, u petom odeljku je prikazana analiza uticaja vrednosti različitih parametara na performanse sistema.

Peto poglavlje sadrži tehnički opis softverske arhitekture sistema koji implementira prototipove modela predstavljenih u disertaciji i daje podršku za izvršavanje eksperimenata opisanih u četvrtom poglavlju.

Šesto poglavlje prikazuje rezultate primene razvijenih modela na nekoliko praktičnih domena uz detaljnu diskusiju postignutih performansi. U prvom, drugom i trećem odeljku šestog poglavlja prikazani su postupci primene modela predloženih u ovoj disertaciji na problem detekcije subjektivnosti, problem višekategorijske klasifikacije i problem pojave novog korisnika u sistemu za davanje preporuka, respektivno.

Sedmo poglavlje zaključuje ovu disertaciju i predstavljena pravce daljeg razvoja.

Zahvaljujem se svim članovima Komisije koji su svojim korisnim sugestijama doprineli da disertacija bude jasnija i preglednija. Posebnu zahvalnost dugujem mentoru prof. dr Aleksandru Kovačeviću i prof. dr Zori Konjović za savete i nesebičnu podršku u toku izrade disertacije.

Veliku zahvalnost dugujem i prof. dr Zoranu Obradoviću i dr Pingu Zhang bez čijih saveta rezultati postignuti tokom izrade disertacije ne bi bili mogući. Takođe, zahvaljujem se dr Felixu Fegeru i dr Irini Koprinska koji su velikodušno podelili detalje svoga rada i obezbedili implementaciju svog modela.

Takođe se zahvaljujem porodici na razumevanju i podršci.

Novi Sad, 2014. Jelena Slivka

1 Uvod

Jedan od najvećih problema sa kojima se danas susrećemo jeste preplavljenost informacijama. Organizacija resursa u predefinisane, semantički smislene kategorije je od presudnog značaja u mnogim praktičnim domenima. Kategorizacija može u značajnoj meri olakšati potragu za resursima smanjenjem fokusa na određenu kategoriju ili skup kategorija. Takođe, topološka struktura podataka može reflektovati suštinske zavisnosti koje postoje među konstitutivnim elementima koji sačinjavaju podatke. Na primer, klasifikacija proteina u familije u okviru ogromnih baza podataka je jedna od glavnih ciljeva istraživanja u oblasti strukturnog i funkcionalnog istraživanja genoma [Enright 2002]. Izuzetno brz rast u količini, ali i kompleksnosti podataka koje je neophodno analizirati je doveo do toga da je automatska klasifikacija podataka postala neobilazan deo mnogih parktičnih sistema kao što su sistemi za detekciju prevare (*fraud detection*) [Kou 2004], sistemi za davanje preporuka (*recommender systems*) [Adomavicius 2005], sistemi za filtriranje spam elektronskih (*e-mail*) poruka [Zhang 2004], itd.

Automatski klasifikacioni modeli se, najčešće, formiraju nadgledanim obučavanjem (*supervised learning*): skup već klasifikovanih dokumenata (obučavajući skup) se koristi za obučavanje klasifikacionog modela. Nakon obuke, klasifikacioni model se koristi za anotaciju novog skupa neklasifikovanih dokumenata (test skup). Kvalitet ovako formiranih klasifikacionih modela u velikoj meri zavisi od kvaliteta korišćenog obučavajućeg skupa. U svrhu kreiranja kvalitetnog modela koji bi imao visoku moć generalizacije, neophodno je kreirati što veći i što raznovrsniji obučavajući skup. Obučavajući skupovi se formiraju od strane eksperata, ručnom anotacijom podataka, što čini pribavljanje nephodne količine anotiranih podataka skupim i dugotrajnim procesom.

Nedostatak anotiranih podataka je moguće ublažiti uključivanjem neanotiranih podataka u proces obučavanja. Zahvaljujući danas prisutnim raznovrsnim sistemima za automatsku kolekciju podataka, lako je i jeftino doći do dovoljne količine podataka koji nisu anotirani. U literaturi mašinskog učenja postoje tri glavne paradigme koje se baziraju na ovom pristupu: polu-nadgledano obučavanje (*semi-supervised learning, SSL*), transduktivno obučavanje (*trunsductive learning*) i aktivno obučavanje[1] (*active learning*) [Freund 1997]. Polu-nadgledano obučavanje se odnosi na metode koje teže da eksploatišu neanotirane podatke za nadgledano obučavanje, pri čemu se instance neanotiranog skupa razlikuju od ciljnih instanci za klasifikaciju (test skupa) ili da eksploatišu anotirane podatke za nenadgledano obučavanje. Metode transduktivnog obučavanja takođe pokušavaju da eksploatišu neanotirane podatke, ali podrazumevajući da su neanotirane instance koje se koriste u

[1] U tekstu disertacije će se za pojam obučavanja modela ravnopravno koristiti termini obučavanje i učenje.

procesu obučavanja takođe i ciljne instance koje je neophodno klasifikovati. Aktivno učenje se odnosi na metode koje među neanotiranim podacima selektuje one najvažnije za anotaciju, nakon čega se te instance prosleđuju anotatoru[2]. Cilj jeste da se minimizuje broj anotiranih instanci neophodnih za obučavanje modela, a da pri tome tačnost rezultujućeg modela bude veća ili jednaka tačnosti koju bi dati model imao ukoliko bi se za obučavanje koristio značajno veći broj slučajno odabranih instanci. [Hady 2008a]. Navedene tehnike u znatnoj meri smanjuju neophodan ljudski rad pri kreiranju obučavajućeg skupa, što je učinilo uspešnu aplikaciju ovih tehnika od velikog interesa i za teoriju i za praksu [Zhu 2008].

U ovoj tezi biće predstavljen pristup koja spada u grupu tehnika polu-nadgledanog obučavanja. Tehnike polu-nadgledanog obučavanja mogu, polazeći od male količine anotiranih podataka i od dovoljno velike količine podataka koji nisu anotirani, proizvesti klasifikatore koji su u rangu ili čak prevazilaze performanse modela kreiranih tehnikama nadgledanog obučavanja na istom skupu podataka. Do danas je dizajnirano više tehnika polu-nadgledanog učenja, međutim njihova primena je limitirana uvođenjem pretpostavki o podacima kojelimitiranju njihovu praktičnu primenu [Chapelle 2006].

Jedna od najuspešnijih tehnika polu-nadgledanog obučavanja za klasifikaciju jeste ko-treining (*co-training*) [Blum 1998]. Ko-trening postavka podrazumeva da na obučavajućem skupu postoji prirodna podela obeležja na dva skupa, koja nazivamo pogledima (*views*). Drugim rečima, svaka instanca skupa podataka može da se opiše uz pomoć dve različite "vrste" informacija čije postojanje nameće ovakvu (prirodnu) podelu obeležja. U ko-trening algoritmu se obučavaju dva različita klasifikatora, svaki koristeći zaseban skup obeležja. Potom, iterativno, svaki od klasifikatora selektuje i anotira neobeležene instance za koje je u stanju da najpouzdanije predvidi klasno obeležje. Ove instance se dodaju u polazni obučavajući skup, nakon čega se klasifikatori ponovo obučavaju na ovako proširenom obučavajućem skupu. Ovaj proces se može posmatrati kao uzajamno obučavanje dva klasifikatora uz pomoć neoznačenih instanci.

Visoke performanse ko-trening algoritma su demonstrirane primenom u mnogim praktičnim domenima gde je poznata prirodna podela obeležja. U [Blum, 1998] ko-trening je primenjen na problem automatske klasifikacije web stranica, pri čemu se kao prvi pogled na web stranicu koriste reči koje se nalaze na samoj web stranici, a kao drugi pogled na web stranicu se koriste reči iz linkova koji ukazuju na datu web stranicu. Ko-trening pristup rešavanju problema razrešavanja višeznačnosti reči (*word sence disambiguation*) je predstavljen u [Yarowsky 1995]. U ovom pristupu je formiran jedan klasifikator

[2] Anotator može biti čovek ili model. U aktivnom učenju se najčešće podrazumeva da je dati annotator jedinstven, da uvek dodeljuje tačnu anotaciju, kao i da cena anotacije ne postoji ili da je barem uniformna za sve instance [Settles 2010].

smisla reči na osnovu lokalnog konteksta reči, a drugi klasifikator na osnovu smisla drugih pojava iste reči u istom dokumentu. U [Riloff 1999] klasifikovane su imeničke fraze (*noun phrazes*) za geografske lokacije time što je posmatrana sama fraza i njen lingvistički kontekst. U [Collins 1999] je izvršena identifikacija imenovanih entiteta (*named entity classification*) korišćenjem načina zapisa reči sa jedne strane i konteksta u kome se ona pojavljuje sa druge strane. Ko-trening je korišćen i za klasifikaciju elektronskih (*e-mail*) poruka [Kiritchenko 2001] na taj način što su reči iz tela poruke tretirane kao prvi pogled, a reči iz subjekta poruke tretirane kao drugi pogled. Filtriranje neželjenih (*spam*) poruka uz pomoć ko-trening algoritma je demonstrirano u [Chan 2004]. U [Hady 2008b] ko-trening je korišćen za prepoznavanje objekata korišćenjem histograma boja i histograma orijentacija koji su ekstrahovani iz dvodimenzionalnih slika. U [Levin 2003] je pomoću ko-treninga unapređena vizuelna detekcija automobila u snimcima video nadzora. U ovoj postavci jedan klasifikator detektuje automobil na osnovu originalnih slika sivog nivoa (*gray-level images*), dok drugi detektuje automobil na slikama sa kojih je uklonjena pozadina. U radovima [Qu 2013; Ghani 2002a] opisano je kako ko-trening može biti od velike koristi u sistemima za davanje preporuka. Ko-trening je primenjen i u mnogim problemima tekstualne kategorizacije kao što su detekcija sentimenata [Wan 2009], prepoznavanje imenskih entiteta (*named-entity recognition*) [Pierce 2001], statističko parsiranje (*statistical parsing*) [Sarkar 2001; Hwa 2003; Steedman 2003], itd.

Međutim, iako je ko-trening moćna parigma, on nije široko primenljiv. Većina skupova podataka koji se sreću u praksi imaju jedinstven skup obeležja, pri čemu ne postoji očigledan ili prirodan način da se definiše podela ovog skupa. Takođe, kako bi ko-trening dostigao maksimum svojih performansi, podela obeležja mora da zadovoljava sledeće uslove: (1) svaki pogled zasebno mora da bude dovoljan za kvalitetnu klasifikaciju (odnosno, ukoliko bi smo raspolagali dovoljno velikim obučavajućim skupom, korišćenje isključivo obeležja jednog od individualnih pogleda u obučavanju bi rezultovalo klasifikatorom dobrih performansi), i (2) dva pogleda moraju da budu uslovno nezavisna pod uslovom da je poznato klasno obeležje primera (odnosno, za svaku instancu skupa podataka, obeležja prvog pogleda nisu u korelaciji sa obeležjima drugog pogleda, osim putem klasnog obeležja). Ukoliko su navedeni uslovi ispunjeni, ciljni koncept koji je moguće naučiti uz postojanje slučajnog šuma je moguće naučiti primenom ko-treninga [Blum, 1998].

Nemogućnost primene ko-treninga u mnogim praktičnim primenama gde bi bio od velike koristi je inspirisala mnoge istraživače da se posvete eliminaciji ili barem relaksaciji teških i nepraktičnih uslova primene ko-treininga. Jedan pravac ovog istraživanja jeste pronalaženje optimalne veštačke[3] podele obeležja

[3] Pod veštačkom podelom se podrazumeva podela obeležja nastala kao rezultat primene određenog metodološkog postupka na skup obeležja. Veštačka podela može, na primer, biti

koja bi garantovala dobre performanse ko-treninga [Feger 2006; Salaheldin 2010; Du 2010]. Međutim, pronalaženje ovakve podele nije trivijalan zadatak – performanse ko-treininga su veoma osetljive na korišćenu podelu obeležja [Nigam 2000b; Muslea 2002]. Takođe, na malim skupovima podataka, kakvim raspolažemo u realnim situacijama gde je ko-trening neophodan, nemoguće je pouzdano verifikovati da li dobijena podela ispunjava zadate uslove [Du, 2010]. Zbog toga, po saznanjima autora, do sada nije pronađena univerzalna metodologija za pronalaženje podele obeležja koja bi garantovala dobre performanse ko-treninga na svim skupovima podataka.

Drugi pristup problemu primene ko-treninga na skupove podataka bez prirodne podele obeležja jesu tehnike koje kombinuju ko-trening algoritam sa tehnikama učenja sa grupom hipoteza (*ensemble learning*) [Zhou 2005a; Li 2007; Hady 2008a]. Ove tehnike se baziraju na zameni dva individualna klasifikatora (definisanih u originalnoj ko-trening postavci) većom grupom klasifikatora. Grupa klasifikatora se eksploatiše radi bolje procene pouzdanosti primera selektovanih za anotaciju i uključivanje u obučavajući skup, u cilju smanjenja šuma koji nastaje dodavanjem pogrešno anotiranih primera u obučavajući skup. Eksperimenti pokazuju da ove metode rezultuju značajnim uvećanjem performansi ko-trening algoritma. Uslovi koji moraju biti zadovoljeni da bi formirana grupa klasifikatora imala dobre performanse jeste da su klasifikatori međusobno različiti (u smislu da prave različite greške prilikom klasifikacije novih instanci) i da svaki od individualnih klasifikatora zasebno ima relativno dobre performanse [Breiman 2001]. Postojeće tehnike koje kombinuju ko-trening i učenje sa grupom hipoteza se baziraju na primeni tehnika učenja sa grupom hipoteza koje raznolikost klasifikatora postižu manipulacijom obučavajučeg skupa (tehnike *bagging* [Breiman 1996] i *boosting* [Freund 1996]) ili manipulacijom ulaznog skupa obeležja (metod slučajnih potprostora, *random subspace method, RSM* [Ho 1998]). Ukoliko je obučavajući skup veoma mali, raznolikost klasifikatora koju je moguće postići primenom ovih tehnika je ograničena [Melville 2003][Kuncheva 2003]. Zbog toga je za primenu ovog pristupa neophodno da inicijalni obučavajući skup bude nešto veći od onog korišćenog u inicijalnoj formulaciji ko-treninga gde se polazi od svega nekoliko anotiranih primera [Blum, 1998]. Ovo može da predstavlja problem u nekim slučajevima gde raspolažemo sa izuzetno malim obučavajućim skupom, npr. u problemu pojave novog korisnika u sistemima za davanje preporuka [Adomavicius 2005].

Po saznanjima autora, do sada nije pronađen univerzalan način da se ko-trening primeni na skupove podataka bez prirodne podele obeležja uz garanciju

---

slučajna podela skupa obeležja na dva pogleda, ili podela nastala kao rezultat optimizacije. neke ciljne funkcije koja rezultuje podelom obeležja koja ispunjava određene uslove koji garantuju dobre performanse ko-treninga.

njegovih performansi i bez povećanja neophodne količine inicijalno anotiranih podataka.

U ovoj tezi predstavljena su dva adaptivna modela za klasifikaciju. Modeli su primenljivi na slučaj kada ne postoji dovoljna količina anotiranih podataka za nadgledano obučavanje. Cilj modela jeste da olakšaju problem mukotrpne ručne anotacije uključivanjem neanotiranih podataka u proces obučavanja. Ovim postupkom želeli bi smo da dostignemo performanse kakve bi imao klasifikator treniran na velikom obučavajućem skupu. Razvijeni modeli baziraju se na primeni ko-trening algoritma. U cilju mogućnosti primene i na skupove podataka koji ne poseduju prirodnu podelu osobina, u razvijenim modelima se ko-trening algoritam kombinuje sa tehnikama učenja sa grupom hipoteza. Za razliku od prethodnih rešenja, gde se unutar ko-treninga formira grupa raznovrsnih klasifikatora eksploatacijom različitih uzoraka inicijalnog anotiranog skupa, razvijeni modeli se baziraju na kreaciji grupe raznovrsnih klasifikatora primenom različitih konfiguracija ko-trening algoritma nad istim inicijalnim anotiranim skupom. Na ovaj način se kreira hijerarhija u grupi klasifikatora: svaki ko-trening klasifikator se zasniva na dva različita klasifikatora, a finalni klasifikator eksploatiše grupu kreiranih ko-trening klasifikatora. Prednost ovog pristupa je u mogućnosti korišćenja značajno manjeg anotiranog obučavajućeg skupa za inicijalizaciju algoritma od postojećih tehnika baziranih na sličnom principu.

Kao različite konfiguracije ko-trening algoritma koriste se različite podele obeležja pomoću kojih se ko-trening primenjuje na skup podataka. Budući da je ko-trening algoritam veoma osetljiv na korišćenu podelu obeležja sa kojom se primenjuje [Nigam 2000b; Muslea 2002], možemo očekivati da ćemo primenom ko-treninga sa m različitih slučajnih podela na istom skupu podataka, kreirati grupu od m međusobno različitih ko-trening klasifikatora. Bazirano na ovome, dva modela predstavljena u ovoj disertaciji kreiraju grupu raznovrsnih klasifikatora primenom različitih slučajnih podela obeležja. Predstavljeni modeli se međusobno razlikuju po načinu kombinovanja dobijenih klasifikatora u cilju davanja finalne predikcije.

Prvi model je nazvan *Algoritam Statistike Slučajnih Podela* (***R**andom **S**plit **S**tatistics **Alg**orithm, RSSalg*). Svaki ko-trening proces rezultuje uvećanim trening skupom koji se sastoji od inicijalno anotiranih instanci i inicijalno neanotiranih instanci koje su anotirane u toku ko-trening procesa. U *RSSalg* se instance ovako nastalih uvećanih obučavajućih skupova integrišu u jedinstveni skup L_{int}, tako što se anotacija svake instance skupa određuje većinskim glasanjem (*majority vote*). Potom se primenjuje genetski algoritam [Goldberg 1989] u svrhu identifikacije i uklanjanja nepouzdano anotiranih instanci iz skupa L_{int}. Konačno, skup L_{int} se koristi za nadgledano obučavanje finalnog klasifikatora kojim se može vršiti predikcija klase novih zapisa koje je neophodno klasifikovati.

Drugi način kombinovanja nezavisno obučenih ko-trening klasifikatora se zasniva na *GMM-MAPML* [Zhang 2011] tehnici estimacije tačnih klasnih obeležja na osnovu višestrukih obeležja pripisanih od strane različitih anotatora nepoznatog kvaliteta (*multiple-annotation setting*). U ovom algoritmu, nazvanom *Integracija Višestrukih Ko-treninranih Klasifikatora* (**I**ntegration of **M**ultiple **C**o-trained **C**lassifiers, *IMCC*), svaki od nezavisno treniranih ko-trening klasifikatora daje predikciju klase za svaku test instancu. U ovoj postavci se svaki od ko-trening klasifikatora tretira kao jedan od anotatora čiji je kvalitet nepoznat, a svakoj test instanci se dodeljuje više klasnih obeležja. Na kraju se primenjuje *GMM-MAPML* tehnika (opisana u odeljku 1.4.3), kako bi se na osnovu dodeljenih višestrukih klasnih obeležja izvršila estimacija stvarnog klasnog obeležja test instance.

Razvijeni modeli evaluirani su primenom postupka unakrsne validacije na 17 skupova podataka različitih po dimezionalnosti, broju instanci i redudantnosti obeležja. Kao mera performanse treniranih klasifikatora korišćena je tačnost (*accuracy*), kao i *F*-mera (*F-measure*). Ovo je široko prihvaćena mera za evaluaciju performansi ko-trening algoritma. U evaluaciji, performanse razvijenih modela poređene su sa performansama ko-treninga primenjenog sa prirodnom podelom obeležja[4], kao i sa alternativnim metodama koje omogućavaju primenu ko-treninga na skupove podataka bez prirodne podele obeležja: ko-trening sa slučajnom podelom obeležja i ko-trening sa veštačkom *maxInd* podelom obeležja predstavljenom u radu [Feger 2008]. Takođe, u cilju provere da li predloženi načini kombinovanja predikcija pojedinačnih ko-trening klasifikatora u formiranoj grupi klasifikatora daju dobre rezultate, performanse *RSSalg* i *IMCC* algoritma su poređene sa performansama dobijenim većinskim glasanjem formirane grupe ko-trening klasifikatora. Konačno, performanse su poređene sa performansama modela treniranog nadgledanim obučavanjem na malom inicijalnom skupu anotiranih podataka, kao i sa performansama istog modela treniranog na mnogo većem obučavajučem skupu koji se sastoji od inicijalnog anotiranog skupa podataka i skupa neanotiranih podataka kojima je za potrebe obuke nadgledanog modela ručno dodeljena tačna anotacija. Poslednji model služi kao dobra aproksimacija ciljnih performansi koje bi smo želeli da dobijemo primenom ko-trening algoritma. Na kraju je radi generalnog poređenja dobijenih klasifikatora na svim skupovima podataka primenjen i *Friedman*-ov statistički test [Friedman 1937; Friedman 1940], praćen post-hoc *Bergmann-Hommel*-ovim testom, što je procedura za poređenje više različitih klasifikatora na više različitih skupova podataka preporučena u [Demšar 2006] i [Garcia 2008]. Takođe, radi analize uspešnosti testiranih algoritama na pojedinačnim skupovima podataka, primenjen je i ANOVA *F*-test.

[4] Na onim skupovima podataka kod kojih data podela prirodno proističe iz postojanja dva odvojena izvora informacija ili je na neki drugi način definisana od strane prethodnih autora.

U izvedenim eksperimentima *IMCC* postavka se pokazala kao generalno bolja od svih poređenih postavki. Na svim testiranim skupovima podataka uspela je da unapredi performanse polaznog klasifikatora treniranog nadgledanim obučavanjem na malom anotiranom skupu podataka, a takođe je dostigla performanse klasifikatora treniranog nadgledanim obučavanjem na značajno uvećanom obučavajućem skupu. Uz idealno odabrane parametre, *RSSalg* postavka dostiže performase *IMCC* postavke. Međutim, nedostatak *RSSalg* postavke jeste uvođenje dodatnih parametara koje je neophodno pažljivo optimizovati. Ovde predložena tehnika optimizacije parametara *RSSalg* postavke se pokazala uspešna samo na redudantnijim skupovima podataka. Ostale testirane alternativne za primenu ko-treninga u slučaju nepoznate prirodne podele obeležja su se pokazale nepouzdane, u smislu da na većini skupova podataka nisu uspele da unaprede performanse polaznog klasifikatora.

Takođe, radi bolje karakterizacije rešenja i testiranja njihove primenljivosti u problemima realnog sveta izvršena je i empirijska analiza osetljivosti rešenja na novo uvedene parametre, kao i na parametre samog ko-treninga. Ispostavilo se da su postavke bazirane na kreiranju grupe ko-trening klasifikatora prilično robustne na korišćeni broj slučajnih podela (odnosno broja klasifikatora u grupi). Inicijalno, sa porastom broja korišćenih podela (tj. broja ko-trening klasifikatora u grupi) rastu i performanse datih postavki, a nakon određene tačke performanse se sa porastom broja klasifikatora više ne menjaju u značajnoj meri (dolazi do konvergencije). Takođe je analiziran način optimizacije parametara *RSSalg* postavke koji direktno utiču na eliminaciju nepouzdano anotiranih instanci iz integrisanog skupa za obuku finalnog klasifikatora. U analizi je utvrđeno da svi korišćeni parametri za eliminaciju nepouzdano anotiranih primera imaju velik uticaj na performanse *RSSalg* postavke. Pogrešan odabir parametara može uzrokovati da *RSSalg* postavka nije u stanju da unapredi ili da čak degradira performanse polaznog klasifikatora. Nije utvrđena generalna preporuka za izbor parametara – ciljna funkcija je veoma osetljiva na podatke tako da ima smisla koristiti genetski algoritam za optimizaciju datih parametara, ali je potrebno pobojšati predloženu funkciju prilagođenosti u cilju dobijanja boljih performansi.

U cilju procene grupe skupova podataka na koje je moguće pouzdano primeniti predstavljene modele meren je i uticaj redudantnosti dobijenih podela na performanse rešenja. Ispostavilo se da je *RSSalg* postavka prilično osetljiva na nedostatak redudantnosti pogleda nastalih slučajnim podelama obeležja. Postavka *RSSalg* sa idealno odabranim parametrima je u značajnoj meri manje osetljiva na ovu osobinu kreiranih podela, dok je *IMCC* postavka, iako osetljiva na redudantnost, donekle robustnija od *RSSalg* postavke.

Što se tiče osetljivosti na same parametre ko-treninga, *RSSalg* postavka sa idealno odabranim parametrima se pokazala kao najrobustnija i na odabir broja iteracija ko-trening algoritma i na odabranu veličinu rasta skupa podataka

u svakoj iteraciji. Sledeća po stabilnosti za broj iteracija je bila postavka sa veštačkom *maxInd* podelom obeležja [Feger 2008], međutim na stabilnost u izvedenim eksperimentima je u značajnoj meri uticao izbor unutrašnjih klasifikatora ko-treninga u *maxInd* postavci.

Konačno, analizirana je mogućnost primene razvijenih modela u nekoliko realnih problema gde nedostatak anotiranih podataka predstavlja veliki problem za zadatak automatske klasifikacije: detekcija subjektivnosti, višekategorijska klasifikacija i problem pojave novog korisnika u sistemu za davanje preporuka. Modeli predstavljeni u disertaciji su uspešno primenjeni na navedene probleme.

U ostatku ovog odeljka biće izložen kratak pregled nadgledanih modela za automatsku klasifikaciju koji se u okviru ove disertacije koriste za formiranje polaznih klasifikatora (odeljak 1.1), genetskih algoritama koji su korišćeni za optimizaciju parametara modela predstavljenih u disertaciji (odeljak 1.2). U odeljku 1.3 je izložen pregled postupaka korišćenih u polu-nadglednom obučavanju. Odeljak 1.4 prezentuje tehnike učenja sa grupom hipoteza blisko povezanih sa modelima predloženim u ovoj disertaciji.

1.1 Nadgledani modeli za automatsku klasifikaciju

U ovom odeljku su predstavljeni modeli koji spadaju u grupu modela formiranih nadgledanim obučavanjem. Ovde predstavljeni modeli su korišćeni u okviru modela predloženih u ovoj disertaciji.

Modelima nadgledanog obučavanja je za učenje potreban skup već klasifikovanih instanci (obučavajući skup). Svaka instanca obučavajućeg skupa je opisana nizom vrednosti zadatih obeležja. Jedno od obeležja je i klasno obeležje koji označava klasu kojoj instanca pripada. Cilj nadgledanog obučavanja jeste da se formira model kojim se klasno obeležje može izraziti kao funkcija vrednosti ostalih obeležja. Ovde će biti izloženi modeli koji su korišćeni za nadgledano obučavanje u okviru ove disertacije.

1.1.1 Naivni Bajesov model za klasifikaciju

Naivni Bajesov model (eng. *naive Bayes*, NB) je jednostavan probabilistički klasifikacioni model koji se oslanja na Bajesovu teoremu.

Neka su date slučajne promenljive X i Y. Označimo sa $P(x) = P(X = x)$ verovatnoću da slučajna promenljiva X ima vrednost x, a sa $P(y) = P(Y = y)$ verovatnoću da slučajna promenljiva Y ima vrednost y. Dalje, označimo sa $P(x,y) = P(X=x,Y=y)$ verovatnoću da će u isto vreme slučajna promenljiva X imati vrednost x, a slučajna promenljiva Y imati vrednost y, a sa $P(x|y) = P(X=x|Y=y)$ verovatnoću da slučajna promenljiva X ima vrednost x ako je poznato da slučajna promenljiva Y ima vrednost y. $P(x,y)$ nazivamo

zajedničkom, a $P(x|y)$ uslovnom verovatnoćom i veza između ove dve vrednosti je sledeća:

$$P(x,y) = P(y|x)\cdot P(x) = P(x|y)\cdot P(y). \quad (1)$$

Na osnovu prethodne jednakosti moguće je izvesti Bajesovu teoremu koja glasi:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}. \quad (2)$$

Ako pretpostavimo da ne postoji deterministička veza između klasnog obeležja i ostalih obeležja, obeležja možemo posmatrati kao slučajne promenljive. Ova pretpostavka je naročito pogodna ukoliko obučavajući skup sadrži greške (npr. dve instance sa istim vrednostima obeležja ali različitom klasom) ili kada postoje neki dodatni, nepoznati faktori koji utiču na klasifikaciju. Označimo klasno obeležje slučajnom promenljivom Y, a ostala obeležja vektorom slučajnih promenljivih $\bar{X}$. Verovatnoća da će instanca sa vrednostima obeležja $\bar{x}$ imati klasu y određena je uslovnom verovatnoćom $P(y|\bar{x})$. Obučavanje klasifikacionog modela se svodi na izračunavanje uslovne verovatnoće $P(y|\bar{x})$ za sve moguće kombinacije vrednosti obeležja $\bar{x} \in \bar{X}$ i sve moguće klase $y \in Y$. Klasifikacija nove instance, koja ima kombinaciju vrednosti obeležja $\bar{x}'$ se svodi na izračunavanje uslovne verovatnoće $P(y|\bar{x}')$ za svaku moguću vrednost $y \in Y$, nakon čega se datoj instanci dodeljuje klasno obeležje y' za koje je dobijena verovatnoća $P(y'|\bar{x}')$ maksimalna.

Verovatnoća $P(\bar{x})$ da data instanca ima određenu kombinaciju vrednosti obeležja $\bar{x} = (x_1, x_2, \dots, x_d)$ je jednaka za sve instance. Zbog ovoga, na osnovu Bajesove teoreme (jednačina 2), sledi da se maksimizacija verovatnoće $P(y|\bar{x})$ svodi na maksimizaciju izraza $P(\bar{x}|y)\cdot P(y)$, gde $P(y)$ predstavlja verovatnoću da instanca pripada klasi y, a $P(\bar{x}|y)$ predstavlja verovatnoću da instanca koji ima klasu y ima vrednosti obeležja $\bar{x}$. Međutim, uslovnu verovatnoću $P(\bar{x}|y)$ je i dalje teško izračunati za sve moguće kombinacije vrednosti obeležja $\bar{X}$. Zbog ovoga se u naivnoj Bajesovoj metodi uvodi pretpostavka da su obeležja instance međusobno statistički nezavisna. Nezavisnost dva obeležja podrazumeva da prisustvo (odnosno, odsustvo) određene vrednosti prvog obeležja ne utiče na prisustvo (odnosno, odsustvo) određene vrednosti drugog obeležja. Ovo je veoma jaka pretpostavka koja često nije ispunjena (odakle i potiče reč '*naivna*' u nazivu), međutim, u praksi se pokazalo da ovaj model rezultuje veoma dobrim performansama. Pretpostavka uslovne nezavisnosti obeležja nam omogućava da uslovnu verovatnoću $P(\bar{x}|y)$ kombinacije vrednosti obeležja izrazimo kao proizvod uslovnih verovatnoća pojedinačnih vrednosti obeležja:

$$P(\bar{x}|y) = \prod_{i=1}^{d} P(x_i|y). \quad (3)$$

Računanje uslovne verovatnoće pojedinačnih obeležja je mnogo jednostavnije. Prednost ovog pristupa je i što ne zahteva ogroman obučavajući skup kakav bi bio potreban za računanje uslovne verovatnoće $P(\bar{x}|y)$, gde bi svaka moguća kombinacija obeležja $\bar{x}$ morala biti reprezentovana radi određivanja verovatnoće.

Uslovnu verovatnoću $P(x_i|y)$ za kategorička obeležja x_i možemo izračunati na sledeći način:

$$P(x_i|y_j) = \frac{n_c}{n}, \tag{4}$$

gde n predstavlja ukupan broj instanci koje pripadaju klasi y_j, a n_c predstavlja broj instanci klase y_j čija je vrednost i-tog obeležja x_i.

U slučaju kontinualnih obeležja možemo primeniti jedan od dva pristupa. Prvi pristup je diskretizacija – kontinualni interval obeležja se deli na podintervale. Svaki podinterval predstavlja vrednost novog, diskretnog (kategoričkog) obeležja. Za vrednost x_i uzima se podinterval u kome se x_i nalazi. Na ovaj način se određivanje uslovne verovatnoće kontinualnog obeležja može svesti na određivanje verovatnoće kategoričkog obeležja.

Drugi način određivanja uslovne verovatnoće kontinualnog obeležja je da pretpostavimo da vrednosti datog kontinualnog obeležja prate neku statističku distribuciju. U ove svrhe se najčešće se koristi Gausova (normalna) distribucija:

$$P(X_i = x_i|Y = y_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} exp\left\{\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}\right\}, \tag{5}$$

gde μ_{ij} i σ_{ij} prestavljaju parametre modela: srednju vrednost i standardnu devijaciju, respektivno. Ovi parametri se mogu estimirati na osnovu vrednosti koje promenljive uzimaju u obučavajućem skupu.

Problem koji se javlja prilikom određivanja uslovne verovatnoće prema formuli 3 jeste što će izračunata uslovna verovatnoća $P(x_i|y)$ biti 0 ukoliko u obučavajućem skupu nije reprezentovana ni jedna instanca klase y koja kao vrednost i-tog obeležja ima x_i. U opisanom slučaju ceo proizvod $P(\bar{x}|y)$ iznosi 0. Do ovakve situacije lako može doći ukoliko je obučavajući skup mali. Ovo je tzv. prebacivanje (*overfitting*). Prebacivanje predstavlja preveliko prilagođavanje modela opserviranim podacima. U ovom slučaju model je previše kompleksan i, iako je njegova greška klasifikacije mala prilikom klasifikacije instanci obučavajućeg skupa, model ima slabu moć generalizacije, odnosno, pravi velike greške prilikom klasifikacije novih opservacija. Ovaj problem je moguće rešiti malim prilagođavanjem formule 3 za izračunavanje uslovne verovatnoće:

$$P(x_i|y_j) = \frac{n_c + mp}{n + m}, \quad (6)$$

gde n predstavlja broj instanci koje pripadaju klasi y_j, n_c broj instanci klase y_j koje kao vrednost i-tog atributa imaju x_i, a m i p su uvedene konstante. Ovo se može posmatrati kao uvođenje dodatnog „virtuelnog“ skupa uzoraka u obučavajući skup. Uvedeni virtuelni skup ima m instanci, a verovatnoća da instanca klase y_j za i-ti atribut ima vrednost x_i je p. Za p se obično uzima $1/k$ gde je k broj vrednosti koje i-ti atribut može da uzme.

Prednost naivnog Bajesovog klasifikatora je njegova robusnost na greške dobijene prikupljanjem podataka ili nedostajuće vrednosti obeležja u obučavajućem skupu. Greške nemaju mnogo uticaja na verovatnoće budući da su one prosečne vrednosti, dok se nedostajući podaci jednostavno ignorišu prilikom izračunavanja verovatnoća. Takođe, naivni Bajesov klasifikator je robustan i na nevažne atribute. Vrednosti nevažnog atributa će biti gotovo uniformno distribuirane po svim klasama, odnosno verovatnoće tog atributa jednako utiču na uslovne verovatnoće svih klasa pa atribut nema uticaja na klasifikaciju.

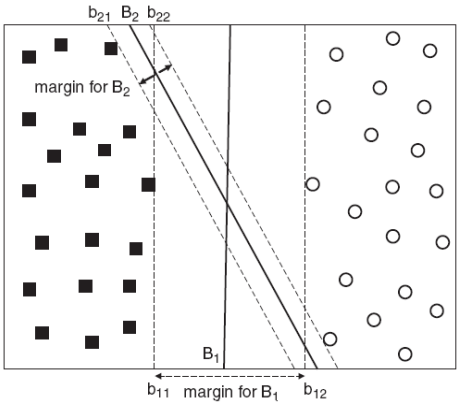
Nedostatak naivnog Bajesa je uvedena pretpostavka o nezavosnosti atributa, zbog čega je ovaj klasifikator osetljiv na korelirane atribute. Atributi koji su u jakoj korelaciji mogu da degradiraju performanse klasifikatora, što se može rešiti uklanjanjem određenih atributa. Detaljniji opis naivnog Bajesovog modela može se naći u [Tan 2005].

1.1.2 Mašine potpornog vektora

Mašine potpornog vektora (*Support Vector Machines*, SVM) [Vapnik 1963] je široko primenljiv klasifikacioni model, posebno pogodan za skupove podataka sa velikim brojem atributa. SVM je linearni klasifikator koji pronalazi hiperravan koja razdvaja dve klase. Prilikom pronalaženja hiperravni, cilj je da margina separacije bude maksimizovana (*maximum margin hyperplane*).

Pojam hiperravni sa maksimalnom marginom separacije je ilustrovan na slici 1. Na slici su prikazane dve moguće hiperravni koje razdvajaju dve klase (reprezentovane kvadratima i krugovima). Obe hiperravni savršeno klasifikuju obučavajući skup (date opservacije) i postavlja se pitanje koja od njih dve će imati bolju mogućnost generalizacije, odnosno, koja će bolje klasifikovati nove (neopservirane) tačke. Posmatrajmo hiperravni B_1 i B_2 sa slike 1. Pomerajmo ravni B_i paralelno na obe strane, dok ne dodirnu jednu od tačaka. Označimo hiperravni dobijene na ovaj način sa b_{i1} i b_{i2}. Margina separacije predstavlja rastojanje između hiperravni b_{i1} i b_{i2}. Kažemo da su podaci linearno separabilni ukoliko postoje ovakve dve hiperravni, paralelne granici odluke, između kojih nema opservacija.

Intuitivno, hiperravan označena kao B_1 na slici 1, sa većom marginom separacije, će imati i veću mogućnost generalizacije – mala pomeranja hiperravni B_2 mogu u značajnoj meri uticati na rezultate klasifikacije, što nije slučaj sa B_1.



Slika 1 Dve moguće hiperravni koje razdvajaju dve linearno separabilne klase (označene krugovima i kvadratima).

Opisani slučaj predstavlja linearni SVM klasifikator koji funkcioniše tako što za zadati obučavajući skup (koji sadrži dve klase) pronalazi hiperravan separacije koja se odlikuje maksimalnom marginom separacije.

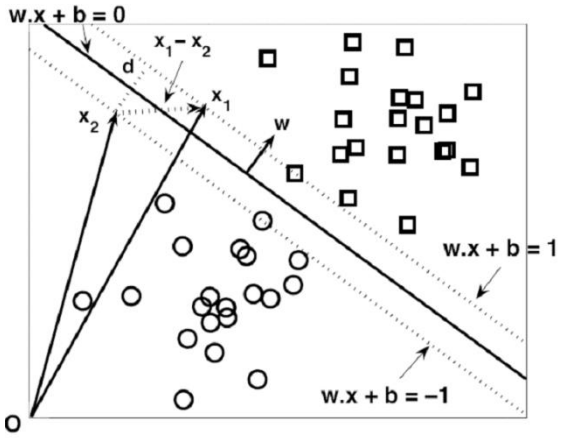
Neka je dat obučavajući skup od *N* instanci klasifikovanih u dve klase označene kao -1 i +1. Označimo *i*-tu instancu sa $(\bar{x}_i, y_i)$ gde $\bar{x}_i = (x_{i1}, x_{i2},..., x_{id})$ predstavlja skup vrednosti obeležja *i*-te instance, a $y_i \in \{1, -1\}$ njeno klasno obeležje. Hiperravan koja razdvaja klase (pomoću koje se vrši klasifikacija) zove se granica odluke (*decision boundary*) i može se predstaviti u sledećoj formi:

$$w*x + b = 0, \tag{7}$$

gde *w* i *b* predstavljaju parametre modela. Svi primeri koji se nalaze na granici odluke moraju da zadovoljavaju jednačinu 7, tako da za dve tačke x_a i x_b koje se nalaze na granici odluke važi:

$$\begin{aligned} w*x_a + b &= 0 \\ w*x_b + b &= 0 \end{aligned} \tag{8}$$

Oduzimanjem druge jednačine od prve dobijamo $w*(x_a - x_b) = 0$, gde je $x_a - x_b$ vektor paralelan granici odluke i usmeren od x_a ka x_b. Pošto je skalarni proizvod vektora $x_a - x_b$ i vektora *w* jednak 0, vektor *w* mora biti normalan na granicu odluke, kao što je prikazano na slici 2.



Slika 2 Ilustracija granice odluke i margine separacije SVM klasifikatora.

Za svaki kvadrat x_s, lociran iznad granice odluke može se pokazati da važi $w*x_s + b = k$, gde je gde je $k > 0$. Slično, za svaki krug ispod granice odluke važi: $w*x_s + b = k'$, gde je $k' < 0$. Ako označimo kvadrate klasom +1 a krugove klasom -1, onda svaku novu tačku z možemo da klasifikujemo na sledeći način:

$$y = \begin{cases} +1, & ako\ je\ w \cdot z + b > 0 \\ -1, & ako\ je\ w \cdot z + b < 0 \end{cases} \tag{9}$$

Granica odluke se može skaliranjem parametara w i b dovesti u takav oblik da za paralelne hiperravno b_{i1} i b_{i2} važi:

$$\begin{aligned} b_{i1} &: w*x + b = 1 \\ b_{i2} &: w*x + b = -1 \end{aligned} \tag{10}$$

Margina granice odluke d je rastojanje između ovih hiperravni. Neka su date dve tačke x_1 i x_2 koje se nalaze na hiperravnima b_{i1} i b_{i2}, respektivno (slika 2). Za ove dve tačke, prema jednačini 10, važi: $w \cdot x_1 + b = 1$ i $w \cdot x_2 + b = 1$. Oduzimanjem ove dve jednačine dobijamo:

$$\begin{aligned} w \cdot (x_1 - x_2) &= 2 \\ \|w\| \times d &= 2 \\ d &= \frac{2}{\|w\|}. \end{aligned} \tag{11}$$

Obučavanje linearnog SVM modela

Obučavanje linearnog SVM svodi se na estimaciju parametara w i b. Parametri moraju da zadovoljavaju jednakosti:

$$\begin{aligned} w \cdot x_i + b &\geq 1 \quad ako\ je\ y_i = 1 \\ w \cdot x_i + b &\leq -1 \quad ako\ je\ y_i = -1 \end{aligned} \tag{12}$$

Ova jednakost implicira da se sve instance obučavajućeg skupa koje pripadaju klasi 1 moraju nalaziti iznad hiperravni $w*x + b = 1$, a da se sve instance obučavajućeg skupa koje pripadaju klasi -1 moraju nalaziti ispod hiperravni $w*x + b = -1$. Kompaktnija forma zapisa ova dva uslova je:

$$y_i \cdot (w \cdot x_i + b) \geq 1, \quad i = 1, \ldots, N, \tag{13}$$

gde N predstavlja broj instanci obučavajućeg skupa. Pored ovih uslova, parametri w i b bi trebali biti takvi da je margina maksimalna. Maksimizacija margine ekvivalentna je mininizaciji funkcije:

$$f(w) = \frac{\|w\|^2}{2}. \tag{14}$$

Budući da je optimizaciona funkcija kvadratna i da su ograničenja linearna po parametrima w i b, ovaj optimizacioni problem je konveksan i može se rešiti primenom metode Lagranžovih množilaca. Prvo, zapišimo ciljnu funkciju tako da uključije i uslove koje rešenja moraju da zadovolje:

$$Lp = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \lambda_i \left(y_i (w \cdot x_i + b) - 1 \right) \tag{15}$$

Ova nova ciljna funkcija je poznata kao Lagranžijan za optimizacioni problem, a parametri λ_i se zovu Lagranžovi množioci. Levi deo Lagranžijana se odnosi na ciljnu funkciju predstavljenu jednačinom 14, a desni se odnosi na nejednakosti predstavljene jednačinama 13. Pretpostavljajući da je $\lambda_i \geq 0$, jasno je da svako rešenje koje narušava ograničenja izražena jednačinama 13 (odnosno, rešenje za koga je $y_i(w \cdot x_i + b) < 1$) uvećava vrednost Lagranžijana. Kako bi smo odredili minimum funkcije Lp za w i b, parcijalne izvode funkcije Lp po w i b izjednačavamo sa 0:

$$\begin{aligned} &\frac{\partial Lp}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^{N} \lambda_i y_i x_i \\ &\frac{\partial Lp}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \lambda_i y_i = 0 \end{aligned} \tag{16}$$

Na osnovu prethodnih jednačina ne možemo odrediti w i b jer su nam nepoznati parametri λ_i. Po definiciji metode za Lagranžove množioce moraju da važe sledeći uslovi (*Khun-Tucker*):

$$\begin{aligned} &\lambda_i \geq 0 \\ &\lambda_i \left[y_i (w \cdot x_i + b) - 1 \right] = 0 \end{aligned} \tag{17}$$

Iz drugog uslova za λ_i u jednačini 17 sledi da za sve vektore x_i za koje važi $y_i(w \cdot x_i + b) = 1$ mora da važi $\lambda_i > 0$, dok za sve ostale vektore x_i mora da važi λ_i

= 0. Obučavajuće instance, odnosno, vektori x_i za koje važi $y_i(w \cdot x_i + b) = 1$ nazivaju se vektori potpore (*support vectors*). Ovi vektori se nalaze na hiperravnima b_{i1} i b_{i2} (hiperravni paralelne granici odluke kojima pripadaju tačke najbliže granici odluke). Pomeranja potpornih vektora rezultuju pomeranjem granice odluke.

Dati optimizacioni problem je i dalje veoma komplikovan za izračunavanje zbog velikog broja parametara: w, b i λ_i. Problem se može pojednostaviti transformacijom Lagranžijana u funkciju koja zavisi isključivo od Lagranžovih množioca. Ovo je poznato kao dualni problem. Kako bi smo ovo uradili, oduzimamo jednačine 16 od jednačine 15 i dobijamo:

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j x_i \cdot x_j \ . \tag{18}$$

Izračunavanjem parametara λ_i, w i b dobijamo jednačinu granice odluke:

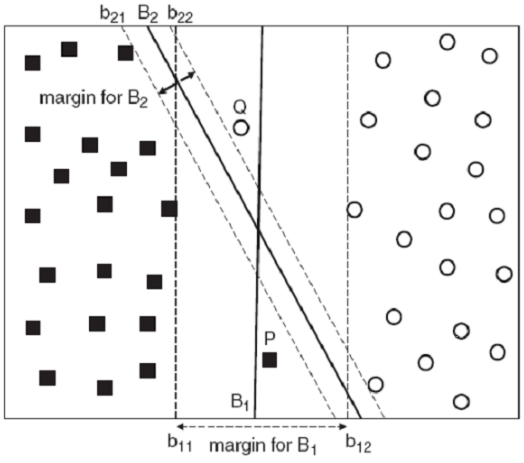
$$\begin{gathered} w \cdot x + b = 0 \\ (\sum_{i=1}^{N} \lambda_i y_i x_i \cdot x) + b = 0 \end{gathered} \ . \tag{19}$$

Sada se nove opservacije mogu klasifikovati pomoću sledeće funkcije:

$$f(z) = sign(w \cdot z + b) = sign\left((\sum_{i=1}^{N} \lambda_i y_i x_i \cdot x) + b \right). \tag{20}$$

Linearni SVM: neseparabilan slučaj

Linearno separabilni podaci predstavljaju idealan slučaj. U realnosti često postoje greške u podacima, npr. nekim instancama može biti dodeljena pogrešna klasa. Linearno neseparabilan slučaj je islustrovan na slici 3.

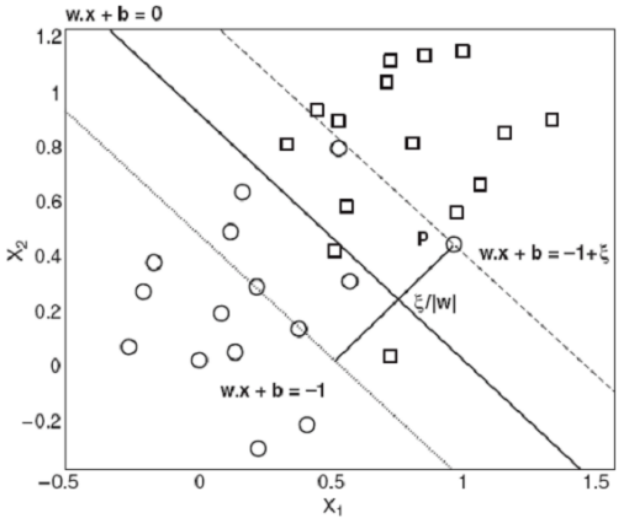


Slika 3 Neseparabilan slučaj za linearni SVM.

U odnosu na sliku 1, na slici 3 dodata su dva nova primera obeležena sa *P* i *Q*. U ovom slučaju hiperravan B_1 ne klasifikuje dobro primere *P* i *Q*, dok ih hiperravan B_2 dobro klasifikuje. Međutim, odabir hiperravni B_2 za granicu odluke bi bio slučaj prebacivanja modela. Bolje je zadržati hiperravan B_1 koja ima širu marginu, budući da tačke *P* i *Q* mogu biti samo šum (greške u podacima). Međutim, prethodno opisanim pristupom ne bismo mogli da izračunamo B_1 jer tačke *P* i *Q* ne bi zadovoljavale uslove iz jednačine 12. Zato je potrebno relaksirate te uslove. Tako dobijamo granicu odluke koja ima relaksiranu marginu (*soft margin*). Novi uslovi imaju sledeći oblik:

$$\begin{aligned} w \cdot x_i + b &\geq 1 - \varepsilon_i \quad ako\ je\ y_i = 1 \\ w \cdot x_i + b &\leq -1 + \varepsilon_i \quad ako\ je\ y_i = -1 \end{aligned} \tag{21}$$

gde je $\varepsilon_i > 0$, $i=1,...,N$, a N predstavlja broj uzoraka. Vrednosti ε_i zovu se promenljive relaksacije (*slack variables*) ili fiktivne promenljive. Na slici 4 je islustrovan SVM klasifikator sa relaksiranom marginom.

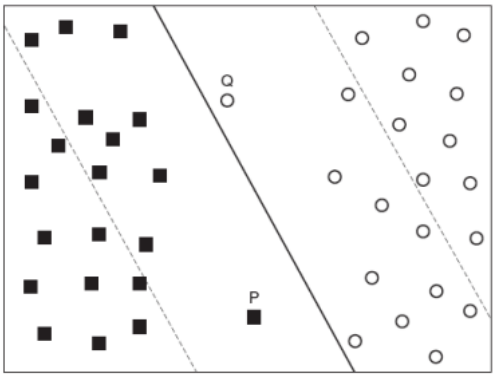


Slika 4 SVM sa relaksiranom marginom.

Ukoliko bi smo uz nove uslove zadržali istu optimizacionu funkciju (jednačina 14), može se desiti da rezultujući SVM ima velike ε_i vrednosti, odnosno široku marginu separacije i, posledično, veliku grešku klasifikacije (slučaj ilustrovan na slici 5).

Kao rešenje ovog problema, uvodi se dodatni parametar *C* koji penalizuje granice odluke sa velikim ε_i vrednostima. Nova funkcija za optimizaciju ima sledeći oblik:

$$f(w) = \frac{\|w\|^2}{2} + C\left(\sum_{i=1}^{N} \varepsilon_i\right)^k, \tag{22}$$



Slika 5 SVM sa previše širokom marginom.

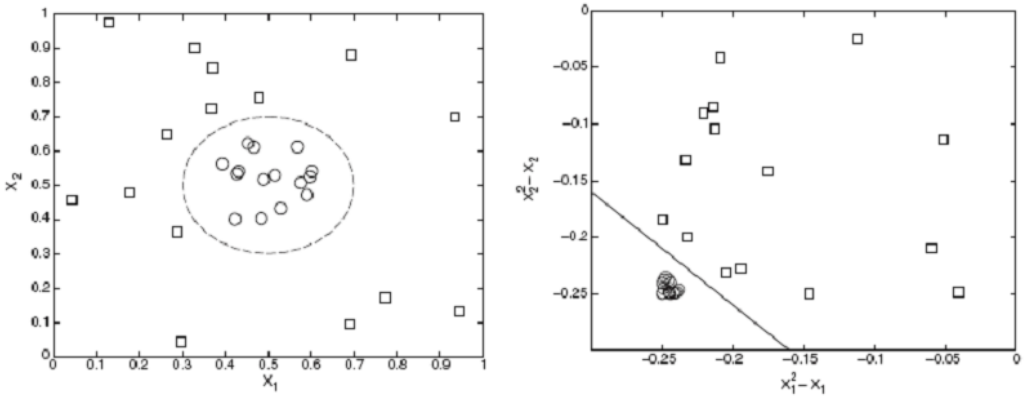
pri čemu su k i C parametri koji zadaje korisnik. Za k=1 izmenjena funkcija za problem Lagranžovih množilaca je:

$$Lp = \frac{1}{2}\|w\|^2 + C\left(\sum_{i=1}^{N}\varepsilon_i\right) - \sum_{i=1}^{N}\lambda_i\left(y_i(w\cdot x_i + b) - 1 + \varepsilon_i\right) - \sum_{i=1}^{N}\mu_i\varepsilon_i \qquad (23)$$

A postupak minimizacije date funkcije je isti kao u prethodnom odeljku. Parametar C je moguće optimizovati na osnovu izdvojenog validacionog skupa.

Nelinearni SVM

Do sada opisani SVM klasifikatori su formirali linearnu granicu separacije između dve linearno separabilne klase. Međutim, podaci ne moraju biti uvek linearno separabilni (slika 6, leva strana). Ovaj problem možemo rešiti transformacijom tačaka iz obučavajućeg skupa u prostor gde su date tačke linearno separabilne, a potom obučiti linearni SVM klasifikator na transformisanom prostoru na način opisan u prethodnim odeljcima (slika 6).



Slika 6 Levo: granica odluke u originalnom prostoru obeležja. Desno: granica odluke u prostoru dobijenom transformacijom.

Označimo sa $\Phi(x)$ funkciju kojom vršimo transformaciju originalnog prostora. Problem obučavanja linearnog SVM u prostoru dobijenom transformacijom možemo zapisati na sledeći način:

$$\min_{w} \frac{\|w\|^2}{2} \quad . \qquad (24)$$
$$y_i \cdot (w \cdot \phi(x_i) + b) \geq 1, \quad i = 1, \ldots, N$$

Dualan problem postaje:

$$L_D = \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i \lambda_j y_i y_j \phi(x_i) \cdot \phi(x_j) . \qquad (25)$$

Problem koji ostaje jeste odabir funkcije $\Phi(x)$ koja će rezultovati linearno separabilnim tačkama u novom prostoru. Takođe, ukoliko transformacija rezultuje prostorom velike dimenzionalnosti, izračunavanje skalarnog proizvoda funkcija $\Phi(x_i) \cdot \Phi(x_j)$ je vremenski i računski veoma zahtevno. Ovi problemi se rešavaju upotrebom tzv. funkcija jezgara K (*kernel function*). Ako pogledamo jednačinu 25, vidimo da nam nije potrebno eksplicitno poznavanje funkcije $\Phi(x)$, već je dovoljno da možemo izračunati skalarni proizvod $\Phi(x_i) \cdot \Phi(x_j)$ za sve tačke originalnog prostora (obučavajućeg skupa). Dakle, želeli bi smo da skalarni proizvod $\Phi(x_i) \cdot \Phi(x_j)$ izrazimo kao funkciju skalarnog proizvoda $x_i \cdot x_j$. Funkcije koje nam daju vezu između skalarnih proizvoda $x_i \cdot x_j$ i $\Phi(x_i) \cdot \Phi(x_j)$ se nazivaju funkcije jezgara (*kernel functions*). Za funkciju jezgara K važi:

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j). \qquad (26)$$

Neke od često korišćenih funkcija jezgra su polinomijalne: $K(x,y)=(x \cdot y)^d$ i $K(x,y)=(x \cdot y + 1)^d$.

Opširniji pregled SVM klasifikatora može se naći u knjizi [Tan 2005].

1.1.3 RBF neuronske mreže

RBF (Radijalne Bazne Funkcije) neuronske mreže su nadgledani obučavajući algoritam predložen u [Moody 1989]. Bazirane su na Koverovoj teoremi o separabilnosti oblika [Cover 1965]: "Verovatnije je da će kompleksan problem klasifikacije oblika biti linearno separabilan ukoliko je nelinearno preslikan u višedimenzionalni prostor, nego u originalnom nižedimenzionalnom prostoru".

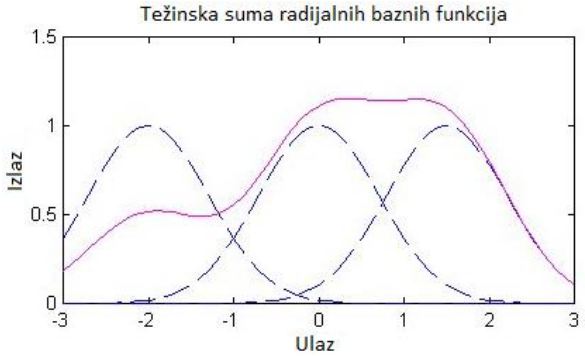
RBF mreža je konceptualno veoma slična modelu K-najbližih suseda. Osnovna ideja je da je da bi predikcija ciljne vrednosti date instance trebala biti bliska predikcijama instanci koje imaju bliske vrednosti varijabli na osnovu kojih se vrši predikcija. U slučaju RBF mreže, u prostor opisan varijablama na osnovu kojih se vrši predikcija se pozicionira jedan ili više RBF neurona (koordinate u kojima se pozicioniraju RBF neuroni se određuju prilikom

obučavanja RBF mreže, o čemu će biti reč kasnije u ovom odeljku) . Zatim se računa Euklidska razdaljina od tačke za koju se vrši predikcija i centra svakog neurona. Na izračunatu razdaljinu se potom primenjuje tzv. radijalna bazna funkcija (takođe poznata kao funkcija jezgra) u cilju da se izračuna (težinski) uticaj svakog neurona. Što je neuron dalje od tačke za koju se vrši predikcija, to je njegov uticaj manji.

Bazne funkcije mogu imati različite forme. Najčešće korišćena je Gausova funkcija:

$$\varphi_j(x_i) = \exp\left(-\frac{\|x_i - c_j\|^2}{\sigma_j^2}\right), \tag{27}$$

gde σ predstavlja standardnu devijaciju (širinu) bazne funkcije, x_i predstavlja ulazni vektor, a c_j centar bazne funkcije. Što je rastojanje između x_i i c_j manje, vrednost j-te bazne funkcije je veća. Predikcija vrednosti za ulazni vektor x_i se vrši težinskim sumiranjem izlaznih vrednosti RBF funkcija (slika 7).

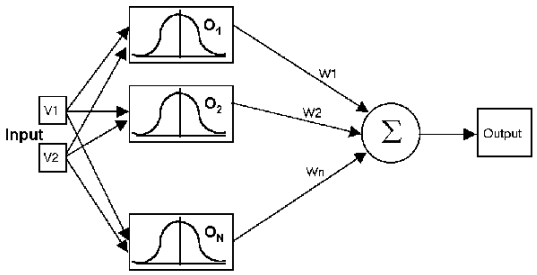


Slika 7 Težinska suma RBF funkcija. Slika je preuzeta sa web sajta http://www.dtreg.com/rbf.htm

RBF se može interpretirati kao neuronska mreža koja se sastoji od tri sloja: ulazni sloj, skriveni sloj i izlazni sloj. U ulaznom sloju postoji po jedan neuron za svaku varijablu na osnovu koje se vrši predikcija. Funkcije transfera neurona u skivenom sloju su nelinearne, a funkcije transfera u izlaznom sloju neurona su linearne. Svaki neuron j skivenog sloja reprezentuje radijalnu-baznu funkciju koja meri rastojanje ulaznog vektora x_i i centra bazne funkcije c_j. Na ovaj način se svaki n-dimenzioni ulazni vektor x_i nelinearno transformiše u m-dimenzioni vektor $\varphi(x_i) = [\varphi_1(x_i), \varphi_2(x_i), ..., \varphi_m(x_i)]$, gde je m broj radijalnih baznih funkcija (skrivenih neurona). Svaki izlazni neuron predstavlja jednu od klasa. Izlazne vrednosti mreže se računaju kao težinska suma aktivacija u skrivenim neuronima:

$$y_k(x_i) = \sum_{j=1}^{m} \varphi_j(x_i) w_{jk}, \tag{28}$$

gde je $y_k(x_i)$ aktivacija izlaznog neurona k, a w_{jk} je težina konekcije skrivenog neurona j i izlaznog neurona k. Arhitektura RBF neuronske mreže je ilustrovana na slici 8.



Slika 8 Arhitektura RBF neuronske mreže. Slika je preuzeta sa web sajta http://www.dtreg.com/rbf.htm

Obučavanje RBF mreže se vrši u dve faze. U prvoj fazi se automatski određuje broj skrivenih neurona m i određuju se parametri radijalnih baznih funkcija c_j i σ_j. Ovo je ključni problem u dizajnu RBF mreža i predstavlja aktivnu oblast istraživanja. Uobičajeni pristup je da se primeni algoritam za klasterizaciju i da se centri c_j podese na vrednosti centroida klastera, a sve širine se σ_j fiksiraju na istu vrednost proporcionalnu maksimalnom rastojanju između određenih centara[5].

Druga faza obuhvata optimizaciju težina na obučavajućem skupu. Ovo se može uraditi minimizacijom sume kvadrata greške. Za vreme ove faze se ne menjaju radijalne bazne funkcije pa je učenje efikasno.

U radu [Park 1991] dokazano je da su RBF mreže univerzalni aproksimatori. Ukoliko je dat dovoljan broj skrivenih neurona sa odgovarajuće odabranim baznim funkcijama, RBF mreže mogu da aproksimiraju bilo koju funkciju proizvoljno dobro. Prednosti RBF mreža jeste njihovo brzo obučavanje, tačna klasifikacija i robustnost na šum.

1.2 Genetski algoritmi

U okviru ove disertacije, za optimizaciju parametara modela su korišćeni genetski algoritmi. Genetski algoritmi (GA) [Holland 1975] su stohastičke heurističke tehnike pretrage kojima se pronalaze približna rešenja kompleksnih optimizacionih problema. GA se obično primenjuju na probleme gde je prostor pretrage velik i kompleksan. U cilju dostizanja optimuma više-dimenzionih problema, genetski algoritmi imitiraju prirodni proces evolucije [Goldberg 1989].

[5] Nije neophodno da sve širine σ_j imaju iste vrednosti. U nekim pristupima se širine neurona određuju nezavisno.

U okviru genetskog algoritma se generiše grupa jedinki (populacija). Svaka jedinka iz populacije predstavlja potencijalno rešenje problema. Rešenja su predstavljena genetskim kodom, najčešće binarnim kodiranjem (niz bitova) koji nazivamo hromozomom (*chromosome*). Kao mera kvaliteta svakog rešenja (jedinke) se koristi definisana funkcija prilagođenosti (*fitness function*). Cilj optimizacije jeste da se pronađe ekstrem funkcije prilagođenosti. Funkcija prilagođenosti se definiše zavisno od domena problema i njen odabir može da utiče na brzinu konvergencije ka rešenju.

Generisanje nove jedinke na osnovu trenutne populacije se naziva reproduckija (*reproduction*). Reprodukcijom se generišu nova potencijalna rešenja koja mogu *naslediti* dobre karakteristike prethodno istraženih potencijalnih rešenja, ali istovremeno omogućiti i pretragu „nepoznate teritorije“. Koje jedinke će učestvovati u reprodukciji se određuje tzv. procesom selekcije (*selection*). Kao i u prirodnom procesu evolucije, favorizuju se kvalitetnije jedinke. U procesu reprodukcije se koriste operatori ukrštanja (*crossover*) i mutacije (*mutation*). Ukrštanjem se vrši rekombinacija gena para jedinki selektovanih za reprodukciju. Razmena genetskog materijala kvalitetnih jedinki omogućava generaciju još kvalitetnijih jedinki. Čak i manje kvalitetne jedinke, sa nekim dobro prilagođenim genima mogu da daju svoj doprinos ukrštanjem sa kvalitetnijim jedinkama. Mutacija predstavlja slučajnu promenu određenog gena jedinke. Primenjuje se u cilju sprečavanja da jedinke postanu međusobo previše slične i kako bi se izbeglo upadanje u lokalne ekstreme funkcije prilagođenosti. Mutacijom se u novu populaciju uvode karakteristike koje jedinke ne bi mogle naslediti od postojećih jedinki. Ovo ubrzava konvergenciju ka rešenju i daje mogućnost da svi delovi prostora rešenja budu pretraženi.

Inicijalna populacija se u genetskom algoritmu obično kreira na slučajan način. Nakon toga se proces reprodukcije primenjuje dok se na osnovu prethodne populacije ne generiše nova populacija iste veličine. Iterativno se, proizvodnjom novih populacija, dolazi do sve boljih rešenja. Postoji više alternativnih kriterijuma za zaustavljanje genetskog algoritma. Neki od njih su: dostignut je maksimalan broj iteracija, dostignut je maksimalan broj evaluacija funkije prilagođenosti, pronađena je jedinka čija funkcija prilagošenosti odgovara unapred zadatoj optimalnoj vrednosti funkcije prilagođenosti, dostignut je maksimalan broj iteracija bez pobojšanja, vremensko ograničenje, itd. Na kraju izvršavanja genetskog algoritma, najbolja jedinka poslednje generacije se proglašava za rešenje problema.

U proces reprodukcije se može uvesti i tzv. elitizam (*elitism*). Naime, zamenom postojeće populacije novom populacijom se može izgubiti do tada najbolje pronađeno rešenje. Čuvanje najboljih jedinki prethodne populacije (koje nepromenjene prelaze u novu populaciju) može u drastičnoj meri ubrzati konvergenciju.

Postoje mnogobrojne prednosti GA. GA se može rešiti svaki optimizacioni problem koji je moguće opisati enkodingom hromozoma. Tehnika pronalaženja optimalnog rešenja nije zavisna od površine funkcije greške, te možemo rešavati multi-dimenzionalne, nediferencijabilne, nekontinualne i čak i neparametarske probleme. GA može da vrati i više rešenja istog problema. Takođe, GA su veoma su laki za razumevanje. Međutim, ne postoji apsolutna garancija da će GA pronaći globalni optimum funkcije prilagođenosti. Takođe, njihova upotreba u aplikacijama koje zahtevaju realno vreme odgovora je ograničena, budući da ne mogu garantovati konstantno vreme optimizacije. Još jedan nedostatak je i postojanje uticaja slučajnosti u dobijenom rešenju.

1.3 Pregled tehnika za polu-nadgledano obučavanje

Tehnike polu-nadgledanog obučavanja su dizajnirane da, pored anotiranih podataka, za obučavanje koriste i neanotirane podatke u cilju povećanja tačnosti i robustnosti algoritama za automatsko zaključivanje u slučajevima kada ne postoji dovoljna količina anotiranih podataka.

Kako neanotirani podaci mogu pomoći procesu obučavanja? Na osnovu neanotiranih podataka možemo dobiti bolju procenu marginalne distribucije $p(x)$ opservacija x (odeljak 1.4.3.1). Da bi polu-nadgledano obučavanje moglo doprineti procesu obučavanja, mora postojati veza između distribucije $p(x)$ i ciljne funkcije mapiranja opservacija x na izlaze y $p(y|x)$ [Seeger 2001]. Ovo očekivanje dovodi do strukturalnih pretpostavki o geometriji podataka. Ukoliko ovo nije slučaj, polu-nadgledano obučavanje može degradirati performanse. Zbog toga, da bi polu-nadgledano obučavanje bilo uspešno moraju biti ispunjeni odgovarajući uslovi. Na primer, jedan od najčešće postavljanih uslova je pretpostavka o glatkoći polu-nadgledanog obučavanja: ukoliko su dva ulaza x_1 i x_2 blizu, onda bi i odgovarajući izlazi y_1 i y_2 trebali biti bliski. Generalno, ne postoji algoritam polu-nadgledanog obučavanja koji je univerzalno superioran u odnosu na ostale. U praksi, potrebno je odabrati model koji uvodi pretpostavke koje najbolje odgovaraju strukturi problema [Chapelle 2006].

U ovom odeljku biće dat kratak pregled postojećih algoritama polu-nadgledanog obučavanja za klasifikaciju, sa naglaskom na onim tehnikama koje su relevantne za ovu disertaciju. Detaljan pregled literature i taksonomije postojećih metoda dat je u [Zhu 2008] i [Chapelle 2006].

1.3.1 Generativni modeli

U statistici, model mešavina predstavlja probabilistički model za reprezentaciju postojanja potpopulacija u celokupnoj populaciji. U ovom modelu se statistička distribucija modeluje sa mešavinom (težinskom sumom) drugih distribucija. Generativni modeli pretpostavljaju model oblika $p(x,y)$ =

$p(y)p(x|y)$, gde $p(x|y)$ predstavlja mešovitu distribuciju koju je moguće identifikovati (odrediti distribucije koje predstavljaju njene komponente).

Primeri generativnih modela koji se često koriste u polu-nadgledanom obučavanju su mešavina Gausovih raspodela (poglavlje 1.4.3.3), mešavina multinominalnih distribucija (Naivni Bajes) [Nigam 2000a] i skriveni Markovljevi modeli (*Hidden markov Models*, *HMM*). Ukoliko raspolažemo velikom količinom opservacija, moguće je identifikovati komponente mešavine nakon čega nam je, u idealnom slučaju, dovoljan jedan anotirani primer po komponenti da bi smo u potpunosti odredili mešovitu distribuciju [Zhu 2008].

Kod generativnih modela potrebno je voditi računa o nekoliko stvari:

- **Mogućnost identifikacije modela mešavina** – model mešavina bi trebao biti takav da ga je moguće identifikovati. Generalno, neka je $\{p_\theta\}$ familija dostribucija indeksirana parametrom θ. Parametar θ je moguće identifikovati ako važi $\theta_1 \neq \theta_2 \Longrightarrow p_{\theta_1} \neq p_{\theta_2}$ do na permutaciju komponenata mešavine. Jednostavan primer modela koji je nemoguće identifikovati jeste model koji predstavlja mešavinu dve uniformne distribucije. Npr. neka je $p(x)$ uniformna raspodela u opsegu [0,1]. Takođe, pretpostavimo da imamo 2 anotirane tačke (0.1, +1) i (0.9, -1). Na osnovu navedenih pretpostavki, ne možemo odrediti anotaciju za tačku $x = 0.5$ jer nemamo načina da se opredelimo za jedan od modela koji bi rezultovali dodelom različitih klasnih obeležja [Zhu 2008]:

$$\mathrm{p(y=1)=0.2, p(x|y=1)=unif(0,0.2), p(x|y=-1)=unif(0.2,1)} \quad (1)$$
$$\mathrm{p(y=1)=0.6, p(x|y=1)=unif(0,0.6), p(x|y=-1)=unif(0.6,1)} \quad (2)$$

  Model (1) sve tačke x iz opsega [0, 0.2] klasifikuje u klasu +1, dok sve tačke x iz opsega [0.2, 1] klasifikuje u klasu -1. Prema ovom modelu, tačka $x = 0.5$ bi bila anotirana klasom -1. Mešavina dve komponente ovog modela – uniformne raspodele na opsegu [0, 0.2] i [0.2, 1] rezultuje distribucijom $p(x)$. Model (2) sve tačke x iz opsega [0, 0.6] klasifikuje u klasu +1, dok sve tačke x iz opsega [0.6, 1] klasifikuje u klasu -1. Prema ovom modelu, tačka $x = 0.5$ bi bila anotirana klasom +1. Mešavina dve komponente ovog modela – uniformne raspodele na opsegu [0, 0.6] i [0.6, 1] takođe rezultuje distribucijom $p(x)$. Naime, čak i ako znamo da je $p(x)$ mešavina dve uniformne distribucije, ne možemo na jedinstven način identifikovati njene komponente.

  Poznato je da je mešavinu Gausijana moguće identifikovati, dok mešavinu multivarijabilnih Bernulijevih raspodela nije [McCallum 1998].
- **Ispravnost modela –** ukoliko je pretpostavljeni model mešavina ispravan, neanotirani podaci će sigurno pobojšati tačnost [Castelli 1995][Castelli 1996][Ratsaby 1995]. Međutim, ukoliko pretpostavljeni model nije ispravan, neanotirani podaci mogu degradirati preformanse [Cozman 2003].
- **EM lokalni ekstrem –** u praksi se komponente mešavine identifikuju primenom metode očekivanje-maksimizacija (*Expectation-Maximization*, *EM*) [Dempster 1977]. Ova metoda je podložna upadanju u lokalni ekstrem koji

može biti udaljen od globalnog maksimuma. Ovaj nedostatak je moguće otkloniti inteligentnim izborom polazne tačke primenom aktivnog učenja (*active learning*) [Nigam 2001].

- **Klasterovanje-i-anotacija** – neki pristupi, umesto korišćenja probabilističkog generativnog modela, primenjuju različite tehnike klasterovanja nad celim skupom podataka, nakon čega anotiraju svaki od klastera uz pomoć anotiranih podataka [Demiriz 1999][Dara 2002]. Ove metode postižu dobre performanse ukoliko pronađeni klasteri odgovaraju pravoj distribuciji u podacima, ali su međutim nezgodni za analizu [Zhu 2008].

Prednosti generativnih modela jesu u njihovom dobro ustanovljenom probabilističkom okviru i u njihovoj velikoj efektivnosti u slučaju kada je pretpostavljeni model približan stvarnom modelu koji opisuje podatke. Međutim, često je teško identifikovati optimalan model za korišćenje, kao i utvrditi korektnost pretpostavljenog modela. Ukoliko pretpostavljeni generativni model ne odgovara podacima, neanotirani podaci mogu povrediti performanse modela umesto da ih unaprede. Takođe, parametri generativnog modela se često procenjuju primenom metode očekivanje-maksimizacija (*Expectation-Maximization*, poglavlje 1.3.3) koja je podložna tome da rezultuje lokalnim ekstremom umesto globalnim.

1.3.2 Samo-obučavanje

Samo-obučavanje (self-training) je tehnika u kojoj se klasifikator obučava na maloj količini anotiranih podataka, nakon čega se tako obučeni klasifikator primenjuje na neanotirane podatke radi njihove klasifikacije. Tipično se najpouzdanije klasifikovani primeri anotiraju i dodaju u obučavajući skup, nakon čega se klasifikator ponovo obučava na uvećanom obučavajućem skupu. Ova procedura se ponavlja predefinisani broj puta. Klasifikator u opisanom procesu koristi sopstvene predikcije da obuči sam sebe. Problem koji postoji jeste što ovim procesom greška u klasifikaciji može da pojačava samu sebe [Zhu 2008].

Samo-obučavanje je uspešno primenjeno na više problema procesiranja teksta prirodnog jezika – određivanje smisla reči (*word-sense disambiguation*) [Yarowsky 1995], identifikacija subjektivnih imenica [Riloff 2003a], određivanje subjektivnosti [Maeireizo 2004], itd. Takođe, u [Rosenberg 2005] samo-obučavanje je primenjeno na problem detekcije objekata gde je pokazalo bolje performanse od do tada najboljeg modela.

1.3.3 Metoda očekivanje-maksimizacija

Metoda očekivanje-maksimizacija (Expectation-Maximization, EM) [Dempster 1977] je iterativna statistička tehnika za estimaciju parametara generativnog modela, primenljiva u slučaju kada se u podacima javljaju

nedostajuće vrednosti. Ukoliko je dat generativni model (npr. naivni Bajesov model) i podaci sa nedostajućim vrednostima, *EM* vrši lokalnu maksimizaciju verodostojnosti parametara i estimaciju nedostajućih vrednosti.

Ovu tehniku je moguće primeniti u polu-nadgledanoj postavci ukoliko se klasna obeležja neanotiranih podataka tretiraju kao nedostajuće vrednosti. Prvi korak jeste da se nadgledanim obučavanjem na osnovu anotiranih instanci odrede parametri generativnog modela. Ovako dobijen model se koristi kako bi se neanotiranim instancama dodelila klasna obeležja. Ovako dodeljenim klasnim obeležjima se dodeljuje i težina koja se računa probabilistički na osnovu očekivanja modela za datu nedostajuću anotaciju. U sledećoj iteraciji se za određivanje parametara modela koriste sve instance (inicijalno anotirane, kao i instance anotirane u prethodnoj iteraciji). Poslednja dva koraka se ponavljaju sve dok dati model ne konvergira (dok nema promene u vrednosti parametara modela). Ovim procesom se istovremeno određuju i parametri modela i nedostajuće anotacije.

Razlika u odnosu na samo-obučavanje jeste što se u samo-obučavanju instance trajno anotiraju i dodaju u obučavajući skup, dok se u *EM* metodi instancama dodeljuju privremene labele koje se mogu menjati u toku *EM* procesa.

Iako *EM* algoritam radi dobro pod uslovima da pretpostavljeni model odgovara podacima (da su ispunjene pretpostavke koje uvodi korišćeni generativni model), narušavanje ove pretpostavke rezultuje lošim performansama [Nigam 2000a].

1.3.4 Ko-trening i učenje na osnovu više pogleda

U ovom odeljku opisan je ko-trening, tehnika polu-nadgledanog obučavanja koja služi kao polazna osnova modela predstavljenih u ovoj tezi.

U originalnoj formulaciji, ko-trening algoritam je moguće primeniti kada skup podataka ima prirodnu podelu obeležja na dva odvojena skupa koje nazivamo pogledima. Na primer, web stranice mogu biti opisane bilo tekstom koji se nalazi na web stranici, bilo tekstom iz hiper-linkova koji ukazuju na datu stranicu. Tradicionalni algoritmi ignorišu postojeću podelu obeležja i sva data obeležja ujedinjuju u jedinstven skup obeležja. Nasuprot tome, ko-trening algoritam eksplicitno koristi podelu obeležja prilikom obučavanja – na istom inicijalnom anotiranom skupu se korišćenjem dva različita pogleda treniraju dva različita klasifikatora. Nakon toga se svaki od klasifikatora primenjuje na skup neanotiranih podataka. Za svaki od klasifikatora, bira se predefinisani broj neanotiranih podataka koji će dati klasifikator anotirati. Biraju se oni primeri za koju dati klasifikator ima najveću pouzdanost klasifikacije. Ovako anotirani primeri se dodaju u skup anotiranih podataka, nakon čega se klasifikatori ponovo obučavaju na tom skupu. Ovaj proces se ponavlja u više iteracija i, pod odgovarajućim uslovima, rezultuje klasifikatorima značajno većih performansi u

odnosu na inicijalni klasifikator izgrađen isključivo na anotiranim podacima. Nakon obuke, klasifikatori se mogu primeniti na primere za koje je neophodno odrediti klasno obeležje tako što se, za svako od mogućih klasnih obeležja, pomnože estimacije verovatnoće različitih klasifikatora da instanca pripadnosti datoj klasi. Instanci se dodeljuje obeležje za koje je na ovaj način izračunata verovatnoća najveća. U algoritmu 1 je predstavljen pseudo kod ko-trening algoritma.

Ulaz

- Mali skup L anotiranih instanci
- Znatno veći skup U primera koji nisu anotirani
- Skup neanotiranih instanci T koje je neophodno razvrstati u pozitivnu i negativnu klasu
- Skup obeležja X kojima su opisani dati skupovi podataka
- Parametri ko-trening algoritma:
  - podela skupa obeležja X na skupove obeležja X_1 i X_2
  - broj iteracija ko-trening algoritma k
  - veličina podskupa neanotiranih primera u'
  - brojevi primera p i n koji će u svakoj iteraciji biti anotirani pozitivnom, odnosno negativnom klasom, respektivno, i dodati u inicijalni obučavajući skup)
  - unutrašnji klasifikatori h_1 i h_2

Izlaz

- Uvećan obučavajući skup L_{res} koji se sastoji od inicijalnog obučavajućeg skupa L i primera anotiranih i dodatih u skup L u toku ko-trening procesa
- Klasifikatori h_1 i h_2 obučeni na uvećanom obučavajućem skupu L_{res}
- Skup $T = \{(x_i, y_i)\}$, $i = 1..n$ gde x_i predstavlja i-tu instancu ulaznog obučavajućeg skupa D, a y_i estimaciju tačnog klasnog obeležja instance x_i

Algoritam

Obučavanje:

Za svako i, i=1..k:

- Kreirati podskup neanotiranih primera U' slučajnim odabirom u' primera skupa U.
- Trenirati klasifikator h_1 korišćenjem obučavajućeg skupa L i skupa osobina X_1.
- Trenirati klasifikator h_2 korišćenjem obučavajućeg skupa L i skupa osobina X_2.
- Dozvoliti klasifikatoru h_1 da anotira p pozitivnih i n negativnih primera skupa U'.
- Dozvoliti klasifikatoru h_2 da anotira da anotira p pozitivnih i n negativnih primera skupa U'.
- Dodati ovako anotirane primere u obučavajući skup L.
- Na slučajan način odabrati $2p + 2n$ primera iz skupa U i prebaciti ih u skup U'.

Klasifikacija novih primera:

Za svaku instancu $t \in T$ verovatnoća da ta instanca pripada kategoriji c_j se računa tako što se pomnože verovatnoće kojom klasifikatori h_1 i h_2 predviđaju da ta instanca pripada kategoriji c_j. Instanci se dodeljuje kategorija kojoj odgovara najveća verovatnoća.

Algoritam 1: Ko-trening algoritam

1.3.4.1 Motivacija za metodologiju ko-trening algoritma

Ko-trening postavka je formalizovana i analzirana u [Blum 1998]. U ovom radu je i teorijski dokazano da ko-trening algoritmi, polazeći od slabog inicijalnog prediktora, mogu učiti koristeći neanotirane podatke ukoliko su određeni uslovi zadovoljeni. Njihova formalna postavka i intuitivni primeri koji daju uvid u funkcionisanje ko-trening algoritma biće ukratko izloženi u ovom odeljku.

Neka postoji prostor instanci $X = X_1 \times X_2$, gde X_1 i X_2 odgovaraju različitim pogledima na instancu, odnosno, svaka instanca x je predstavljena kao par (x_1, x_2). Jedna od pretpostavki važna za uspešnost ko-treninga jeste da svaki od pogleda ponaosob mora biti dovoljan za korektnu klasifikaciju. Odnosno, ukoliko bi smo imali dovoljno velik obučavajući skup, korišćenje samo jednog od pogleda bi rezultovalo klasifikatorom dobrih performansi. Ovo znači da za većinu instanci skupa podataka, klasifikatori izgrađeni na zasebnim skupovima podataka moraju davati istu predikciju klasnog obeležja. Formalnije, neka je D distribucija skupa X i neka su C_1 i C_2 klase koncepata definisane nad X_1 i X_2, respektivno. Pretpostavka koju uvodimo jeste da su sve labele instanci (čija je verovatnoća u distribuciji D veća od 0) konzistentne sa nekom ciljnom funkcijom $f_1 \in C_1$, ali takođe i sa nekom ciljnom funckijom $f_2 \in C_2$. Odnosno, ako f označava kombinovani ciljni koncept nad celim primerom iz X, tada za svaku instancu $x = (x_1, x_2)$ uočenu sa klasnim obeležjem l važi $f(x) = f_1(x_1) = f_2(x_2) = l$. Ovo znači da distribucija D dodeljuje verovatnoću 0 svakoj instanci za koju važi $f_1(x_1) \neq f_2(x_2)$ i da su koncepti f_1 i f_2 *kompatibilni* sa ciljnom distribucijom D.

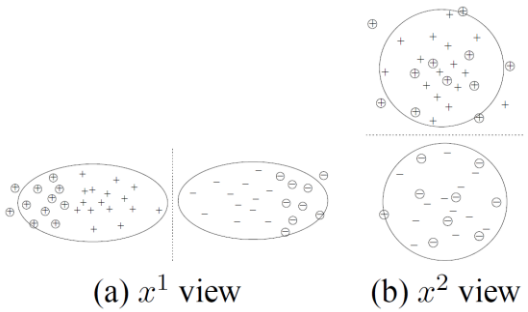
Zahvaljujući ovoj činjenici, možemo se nadati da neanotirane instance mogu pomoći procesu obučavanja. Naime, čak i ako su C_1 i C_2 velike i kompleksne klase koncepata, za datu distribuciju D, skup *kompatibilnih* koncepata bi mogao biti mnogo manji i jednostavniji. Zbog toga se možemo nadati da bi pomoću neanotiranih instanci mogli dobiti bolji uvid u to koji su ciljni koncepti kompatibilni, redukujući broj anotiranih instanci neophodnih za obučavanje.

Za ilustraciju ove ideje može poslužiti sledeći primer – neka je $X_1 = X_2 = \{0, 1\}^n$ i neka su C_1 i C_2 = "konjunkcije nad $\{0, 1\}^n$". Neka je poznato da je prva koordinata relevantna ciljnom konceptu f_1 (npr. ukoliko je prva koordinata $x_1 = 0$, tada važi da je $f_1(x_1) = 0$ pošto je f_1 konjunkcija). Tada, svaka anotirana instanca (x_1, x_2) kod koje je prva koordinata x_1 jednaka 0 može da bude iskorišćena kao negativno anotirana instanca x_2 za concept f_2. Uspešnost ovoga zavisi i od distribucije D. Naime, ukoliko je D takva distribucija gde su x_1 i x_2 veoma korelirani (npr. D daje verovatnoću veću od nule samo instancama za koje važi da je $x_1 = x_2$), tada nam ovo ne daje nikakve korisne informacije o f_2. Međutim, ukoliko su x_1 i x_2 uslovno nezavisni u odnosu na labelu, ukoliko x_1 ima za prvu komponetu 0, x_2 predstavlja *slučajanu* negativnu instancu za f_2, što može biti veoma korisno. Ovo nas dovodi do druge pretpostavke važne za

uspešnost ko-trening algoritma – u idealnom slučaju, obeležja jednog pogleda bi trebala biti uslovno nezavisna od obeležja drugog pogleda, ukoliko je poznato klasno obeležje:

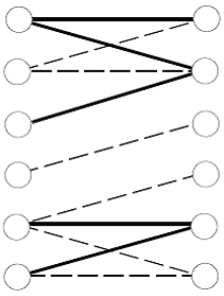
$$Pr_{(x_1,x_2)\in D}[x_1 = \widehat{x_1}|x_2 = \widehat{x_2}] = Pr_{(x_1,x_2)\in D}[x_1 = \widehat{x_1}|f_2(x_2) = f_2(\widehat{x_2})]$$
$$Pr_{(x_1,x_2)\in D}[x_2 = \widehat{x_2}|x_1 = \widehat{x_1}] = Pr_{(x_1,x_2)\in D}[x_2 = \widehat{x_2}|f_1(x_1) = f_1(\widehat{x_1})] \quad (29)$$

Vizuelizacija ove pretpostavke je data na slici 9, preuzetoj iz [Zhu 2008].



Slika 9 Ko-trening pretpostavka o uslovnoj nezavisnosti pogleda. Prema ovoj pretpostavci, instance visoke pouzdanosti u pogledu x^1, reprezentovane zaokruženim labelama, biće slučajno rasute u pogledu x^2. Ovakve instance su u pogledu x^2 veoma informativne jer su nezavisno distribuirane.

U [Blum 1998] ideja ko-treninga je ilustrovana na primeru težinskog bipartintnog grafa. Neka se distribucija D može predstaviti kao bipartitni graf $G_D(X_1,X_2)$. Leva strana grafa ima po jedan čvor za svaku tačku iz X_1, a desna strana ima po jedan čvor za svaku tačku iz X_2. Grana (x_1, x_2) postoji samo ukoliko primer (x_1, x_2) ima verovatnoću veću od 0 u distribuciji D, pri čemu je težina grane jednaka verovatnoći datog primera. U ovoj reprezentaciji, potpuno "kompatibilni" koncepti odgovaraju particiji grafa koja ne preseca ni jednu granu. Ilustracija opisanog grafa se nalazi na slici 10 preuzetoj iz [Blum 1998].



Slika 10 Graf G_D. Grane grafa predstavljaju primere koji imaju verovatnoću pojave veću od nule u distribuciji D. Grane predstavljene punom linijom predstavljaju opservacije nekog uzorka S. Pod navedenom pretpostavkom o kompatibilnosti ciljnih funkcija, čak i bez uočavanja klasnih obeležja, obučavajući algoritam može da zaključi da dva čvora koja pripadaju istoj povezanoj komponenti u grafu moraju imati isto klasno obeležje.

1.3.5 Izbegavanje promena u gustim regijama

Mnogi modeli za polu-nadgledano obučavanje uvode pretpostavku da granica odluke (*decision boundary*) ne treba da prolazi kroz regije u kojima je $p(x)$ visoko. U ovakve modele spadaju transduktivne mašine potpornih vektora (*transdiuctive support vector machines*, *TSVMs*) [Joachims 1999], gausovski procesi (*gaussian processes*) [Lawrence 2005], regularizacija informacija (*information regularization*) [Szummer 2002], minimizacija entropije (*entropy minimization*) [Zhu 2008]. Kao i kod ostalih polu-nadgledanih modela, ukoliko uvedene pretpostavke nisu zadovoljene, ove metode će imati slabije performanse. Na primer, ovo se može dogoditi ukoliko su podaci generisani od strane dve snažno preklapajuće Gausove distribucije, zbog čega bi granica odluke trebala da prolazi upravo kroz najgušću oblast.

1.3.6 Modeli polu-nadgledanog obučavanja bazirani na grafovima

Graf-bazirani modeli polu-nadgledanog obučavanja generišu graf čiji čvorovi odgovaraju anotiranim i neanotiranim instancama skupa podataka, a grane (kojima može biti dodeljena težina) reflektuju sličnost instanci. Ove metode se obično baziraju na pretpostavci o „glatkoći“ klasnih obeležja u grafu [Zhu 2008].

1.3.7 Korišćenje znanja o proporcijama klasa

Poznavanje proporcija klasa neanotiranih primera može biti važno za polu-nagledano obučavanje [Zhu 2008]. Pod proporcijama klasa ovde se misli na proporcije instanci klasifikovanih u svaku od klasa (npr. 20% pozitivnih i 80% negativnih). Bez uvođenja ograničenja o proporcijama, metode polu-nagledanog obučavanja imaju tendenciju da proizvode nebalansirani izlaz, a u ekstremnom slučaju se može desiti da se svi neanotirani podaci svrstaju u istu klasu, što je veoma nepoželjno.

Zbog toga, mnoge metode polu-nadgledanog obučavanja koriste neku formu ograničenja nad klasnim proporcijama. Željene klase propocija se ili daju kao ulaz u algoritam ili se procenjuju na osnovu proporcija klasa anotiranog skupa podataka.

1.3.8 Učenje efikasnog enkodiranja domena na osnovu neanotiranih podataka

Na osnovu neanotiranih podataka moguće je naučiti efikasno enkodiranje obeležja problemskog domena. Anotirani podaci se potom reprezentuju korišćenjem dobijenog enkodinga, a klasifikacija se potom vrši primenom standardnog nadgledanog obučavanja. Jedan primer jeste primena PCA metode (*Principal Component Analysis*) za redukciju dimenzionalnosti

korišćenjem neanotiranih podatakaa, nakon čega se za anotirane podatke koristi rezultujuća niže-dimenzina reprezentacija.

1.3.9 Odnos polu-nadgledanog obučavanja sa sličnim oblastima

U ovom odeljku će biti dat kratak pregled oblasti blisko povezanih sa polu-nadgledanim obučavanjem.

- **Spektralno klasterovanje** (*spectral clustering*) – cilj spektralnog klasterovanja jeste da se čvorovi datog grafa balansirano particionišu u grupe na taj način da se minimizira broj (ili ukupna težina) presečenih grana. Veoma slične spektralnom klasterovanju su graf-bazirane metode polu-nedgledanog obučavanja koje takođe konstruišu graf od podataka. Međutim, za razliku od spektralnog klasterovanja koje je u potpunosti nenadgledan proces baziran samo na težinama grafa, polu-nadgledano obučavanje koristi i anotirane podatke i cilj mu je, u opštem slučaju, anotacija podataka [Narayanan 2007].
- **Učenje sa pozitivnim i neanotiranim podacima** – u mnogim praktičnim aplikacijama dostupni anotirani podaci pripadaju samo jednoj od dve moguće kategorije. Sa druge strane, neanotirani podaci sadrže obe kategorije. Ovo je specifična metoda polu-nadgledanog obučavanja kojoj je cilj da se u neanotiranom skupu podataka identifikiju pozitivni dokumenti [Zhu 2008].
- **Polu-nadgledana klasterizacija** – ovo je oblast veoma slična polu-nadgledanom obućavanju – cilj je klasterizacija, ali koriste se i anotirani podaci u formi *obavezni-linkovi* (restrikcija da dve tače moraju pripadati istom klasteru) i *zabranjeni-linkovi* (dve tačke ne smeju da pripadaju istom klasteru) [Zhu 2008].
- **Polu-nadgledana regresija** – većina graf-baziranih metoda polu-nadgledanog obučavanja estimira „meke labele“ (*soft labels*) pre izvršavanja klasifikacije. Odnosno, estimira se funkcija koja teži da bude bliska ciljevima *y* u anotiranom skupu, a da istovremeno bude glatka u odnosu na graf. Zbog toga, većina graf-baziranih polu-nadgledanih metoda može takođe prirodno da vrši i regresiju [Yhu 2008]. Predložena je i ko-trening postavka za polu-nadgledanu regresiju [Zhou 2005b].
- **Aktivno učenje** (*active learning*) – aktivno učenje i polu-nadgledano obučavanje se suočavaju sa istim problemom – neanotirani podaci su retki i do njih se teško dolazi. Međutim, ove dve oblasti napadaju isti problem iz različitih smerova – dok polu-nadgledano obučavanje eksploatiše ono što obučavajući algoritam misli da zna o neanotiranim podacima, metodi aktivnog učenja pokušavaju da istraže nepoznate aspekte [Settles 2010]. Ovi pristupi se lako i prirodno kombinuju. Neki primeri kombinovanja ko-trening algoritma sa aktivnim učenjem su [Muslea 2002][Munkhdalai 2012][Zhang 2014].

- **Nelinearna redukcija dimenzionalniosti** – cilj ovih metoda jeste da pronađu verno nisko-dimenziono mapiranje podataka visoke dimenzionalnosti. Pripadaju nenadgledanim metodama obučavanja, međutim, način na koji funkcionišu je veoma sličan graf-baziranom polu-nadgledanom obučavanju [Zhu 2008].
- **Učenje metrike rastojanja –** mnogi algoritmni zavise od metrike rastojanja (sličnosti) definisane nad instancama *X*. Ovaj proces je potpomognut neanotiranim podacima. Na primer, modeli graf-baziranog polu-nadgledanog obučavanja se oslanjaju na ove metode. Detaljnije o ovim metodama se može naći u [Zhu 2008].
- **Pronalaženje mehanizma semplovanja instanci za anotaciju** - Mnogi algoritmi polu-nadgledanog obučavanja podrazumevaju da su instance skupova *L* i *U* semplovane nezavisno iz iste raspodele (skraćeno *i.i.d. – independent and identically distributed*). Međutim, ovo nije uvek slučaj [Rosset 2005]. Na primer, neka je ciljna varijabla *y* binarna vrednost koja označava zadovoljstvo mušterije koja je učestvovala u anketi. Međutim, često samo učešće u anketi (čime i sami anotirani podaci) zavise od satisfakcije *y*.
- **Selekcija modela bazirana na metrici** [Schuurmans 2001] – ovo je metoda detekcije nekonzistentnosti hipoteze pomoću neanotiranih podataka. Na primer, možemo imati dve hipoteze konzistnentne nad *L* (sa greškom klasifikacije 0) koje mogu biti nekonzistentne na mnogo većem skupu neanotiranih podataka *U*. Ukoliko je to slučaj, trebali bi odabrati bolju, npr. onu manje kompleksnu.
- **Učenje sa više instanci** (*multi-instance learning*) – u slučaju učenja sa više instanci obučavajući skup se sastoji od anotiranih grupa, gde se svaka grupa sastoji od mnogo neanotiranih instanci. Grupa je anotirana pozitivnom klasom ukoliko sadrži barem jednu pozitivnu instancu, a anotirana negativnom klasom ukoliko sve instance unutar nje pripadaju negativnoj klasi. U [Zhou 2007a] je pokazano da pod pretpostavkom da su instance *i.i.d.* semplovane, učenje sa više instanci postaje specijalan slučaj polu-nadgledanog obučavanja.

1.4 Pregled tehnika učenja sa grupom hipoteza

Pod grupom klasifikatora (*ensemble*) podrazumeva se skup individualnih prediktora (npr. neuralnih mreža ili stabala odlučivanja) čije se predikcije kombinuju u svrhu klasifikacije date instance. U [Dietterich 2000] objašnjeni su fundamentalni razlozi zbog kojih je grupa klasifikatora obično tačnija od svojih individualnih članova. Od presudnog značaja za uspeh grupe jesu različitost grešaka i tačnost pojedinačnih klasifikatora grupe [Breiman 2001]. Pod različitošću klasifikatora se podrazumeva da klasifikatori prave različite greške na datom skupu instanci koje je potrebno klasifikovati. Tehnike učenja sa grupom hipoteza se međusobno razlikuju upravo po načinu postizanja različitosti među klasifikatorima grupe: manipulacijama obučavajućeg skupa

[Breiman 1996; Freund 1996], manipulacijama skupa obeležja [Ho 1998], manipulacijom izlaznih ciljeva [Dietterich 1991] i unošenjem slučajnosti (*randomness injection*). U ovom odeljku biće dat pregled tehnika učenja sa grupom hipoteza koje su od značaja za ovu tezu.

1.4.1 *Bagging* tehnika

Neka je dat obučavajući skup L veličine m i neka je zadat obučavajući algoritam koji ćemo označiti sa *BaseLearn*. U *bagging* tehnici [Breiman 1996] kreira se skup od N klasifikatora h_i ($i = 1 \ldots N$). Svaki klasifikator h_i dobija se primenom obučavajućeg algoritma *BaseLearn* na tzv. samorazvijajuću (*bootstrap*) repliku originalnog skupa podataka veličine m koja se dobija slučajnim uzorkovanjem sa zamenom instanci originalnog skupa podataka. Svaka replika skupa podataka dobijena na ovaj način sadrži oko 63% originalnog skupa podataka [Breiman 1996], pri čemu se jedna instanca može pojaviti više puta u istoj replici.

Finalna hipoteza $H(x)$ dobijene grupe klasifikatora za instancu x se dobija uprosečavanjem distribucije verovatnoća pojedinačnih članova grupe $P_i(x)$, nakon čega se odabira klasa za koju je ovako dobijena verovatnoća najveća:

$$H(x) = argmax_{1 \le k \le K} P(x), \text{ gde je } P(x) = \frac{1}{N} \sum_{i=1}^{N} P_i(x). \tag{30}$$

Ova tehnika pokazuje dobre performanse u slučajevima gde je obučavajući algoritam nestabilan, odnosno gde male promene u obučavajućem skupu mogu dovesti do velikih promena u izlaznoj hipotezi.

1.4.2 Metod slučajnih potprostora

Metod slučajnih potprostora (*random subspace method, RSM*) [Ho 1998] je tehnika učenja sa grupom hipoteza koja raznolikost klasifikatora proizvodi manipulacijom skupa obeležja.

Neka je skup podataka predstavljen sa D obeležja. Slučajno izabranih d obeležja tog skupa predstavljaju d-dimenzioni slučajni potprostor originalnog skupa od D obeležja. U *RSM* metodi se za svaki kreirani slučajni potprostor obučava poseban klasifikator. Finalna predikcija se formira uprosečavanjem distribucija verovatnoća pojedinačnih klasifikatora grupe $P_i(x)$, nakon čega se instanci dodeljuje kategorija za koju je tako dobijena verovatnoća najveća.

1.4.3 *GMM-MAPML* algoritam

Prilikom nadgledanog obučavanja se obično podrazumeva da su instance obučavajućeg skupa anotirane jedinstvenom tačnom labelom. Međutim, u praksi se često srećemo i sa situacijama gde je tačna labela instance nepoznata, a dostupna su višestruka klasna obeležja dodeljena od strane više anotatora (ili nekih drugih izvora) i među dodeljenim klasnim obeležjima postoji neslaganje.

Na primer, u oblasti kompjuterski potpomognutih dijagnoza (*Computer Aided Diagnosis, CAD*), zlatni standard (da li je sumnjiva regija maligna ili nije) je moguće potvrditi samo biopsijom tkiva, što predstavlja invazivnu i skupu procedure. Zbog toga se *CAD* sistemi grade na osnovu klasnih obeležja dodeljenih od strane više radiologa koji daju subjektivnu procenu koja može sadržati šum. U opisanoj situaciji moguće je primeniti *GMM-MAPML* algoritam [Zhang 2011], razvijen sa ciljem estimacije pravog klasnog obeležja na osnovu višestrukih obeležja dodeljenih od strane nezavisnih anotatora nepoznatog kvaliteta. Pored nepouzdanosti anotatora, *GMM-MAPML* modeluje i činjenicu da tačnost anotatora može zavisiti i od samih podataka koje je neophodno anotirati, odnosno, da isti anotator može biti nekonzistentno tačan na različitim grupama podataka.

Neka je dat skup podataka D u kome je svaka instanca anotirana od strane R anotatora, odnosno $D=\{x_i, y_i^1, \ldots, y_i^R\}$, gde x_i predstavlja i-tu instancu skupa D, a $y_i^j \in \{0,1\}$ predstavlja binarno klasno obeležje instance x_i, dodeljeno od strane j-og anotatora.

Prvi korak u *GMM-MAPML* algoritmu jeste primena metoda očekivanje-maksimizacija (*Expectation-Maximization, EM*) i Bajesovog informacionog kriterijuma (*Bayesian information criterion, BIC*) zarad estimacije parametara modela Gausovih mešavina (*Gaussian Mixture Model, GMM*) kako bi dati model odgovarao distribuciji instanci obučavajućeg skupa. Rezultat ovog postupka su određene komponente Gausove mešavine i odgovornosti τ_{ik} (za instancu x_i, τ_{ik} predstavlja verovatnoću da Gausova komponenta k generiše datu instancu, odnosno verovatnoća da instanca x_i pripada k-tom klasteru). Ovaj postupak je objašnjen u odeljku 1.4.3.3.

Nakon što su instance (putem odgovornosti τ_{ik}) klasterovane u grupe međusobno sličinih instanci, bazirano na intuiciji da realni anotatori imaju različitu senzitivnost (*sensitivity*) i specifičnost (*specificity*) za različite grupe instanci, definišemo senzitivnost α_k^j i specifičnost β_k^j j-tog anotatora ($j \in \{1..R\}$) za k-tu Gausovu komponentu ($k \in \{1..K\}$) na sledeći način:

$$\alpha_k^j = \Pr\left(y_i^j = 1 \mid y_i = 1 \; i \; Gausova \; komponenta \; k \; generiše \; x_i\right) \quad (31)$$

$$\beta_k^j = \Pr\left(y_i^j = 0 \mid y_i = 0 \; i \; Gausova \; komponenta \; k \; generiše \; x_i\right) \quad (32)$$

GMM-MAPML pretpostavlja da anotatori generišu predikcije klasnog obeležja na sledeći način: polazeći od instance x_i kojoj je neophodno dodeliti klasno obeležje, anotator pronalazi komponentu Gausovih mešavina za koju je najverovatnije da je generisala instancu x_i. Zatim anotator generiše klasno obeležje na osnovu svoje senzitivnosti i specifičnosti za datu Gausovu komponentu.

Inicijalizacija probabilističkih labela z_i (tj. verovatnoću da je tačna labela 1) se vrši pomoću metode većinskog glasanja. Nakon toga, algoritam

naizmenično izvršava estimaciju maksimalne verodostojnosti (*Maximum Likelihood*, *ML*) i estimaciju maksimalne apostorierne vrednosti (*Maximum a Posteriory Probability*, *MAP*): na osnovu trenutne estimacije probabilističkih labela z_i, *ML* estimacija meri performanse anotatora (njihovu senzitivnost α i specifičnost β) za svaku komponentu i određuje parametre datog klasifikatora w; na osnovu procenjenih vrednosti za senzitivnost α, specifičnost β i parametara klasifikatora w, *MAP* estimacija primenom Bajesovog pravila ažurira probabilističke labele z_i. Nakon što ove dve estimacije konvergiraju, *GMM-MAPML* algoritam kao izlaz daje i estimirane labele instanci z_i i parametre modela $\phi = \{w, \alpha, \beta\}$. Estimacije skrivenih tačnih labela u *GMM-MAPML* algoritmu zavise kako od labela svih anotatora, tako i od samih opservacija.

1.4.3.1 ML estimacija parametara modela

Makimalna veordostojnost (*Maximum Likelihood*, *ML*) se koristi prilikom estimacije parametara statističkog modela. Primenom ove metode se, polazeći od fiksiranog skupa podataka i zadatog statističkog modela, selektuju one vrednosti parametara modela koje maksimizuju funkciju očekivanja. Ovim se zapravo maksimizuje slaganje pretpostavljenog modela sa opserviranim podacima.

Neka su dati skup podataka D i trenutne estimacije skrivenih labela instanci skupa D, označene sa z_i. U ML koraku se maksimizacijom uslovne verovatnoće estimiraju parametri modela $\phi = \{w, \alpha, \beta\}$.

Neka je $z_{ik} = z_i \cdot \tau_{ik}$, gde τ_{ik} predstavlja verovatnoću da je instanca x_i generisana od strane komponente k. Primenom formula (31) i (32) možemo dobiti senzitivnost j-tog anotatora za k-tu komponentu:

$$\alpha_k^j = \sum_{i=1}^{N} z_{ik} y_i^j \Big/ \sum_{i=1}^{N} z_{ik} \tag{33}$$

$$\beta_k^j = \sum_{i=1}^{N} (\tau_{ik} - z_{ik})(1 - y_i^j) \Big/ \sum_{i=1}^{N} (\tau_{ik} - z_{ik}) \tag{34}$$

Primenom ML estimacije moguće je odrediti parametre w za bilo koji model. U radu [Zhang 2011] dat je primer obučavanja modela logističke regresije primenom *Newton-Raphson* metode [Bishop 2006]. Nakon određivanja parametara modela možemo izračunati a priori verovatnoću p_i pozitivne klase:

$$p_i = PR[y_i = 1 | x_i, w] = \sigma(w^T x_i) \tag{35}$$

gde σ predstavlja logističku sigmoidnu funkciju $\sigma(x) = 1/(1 + e^{-x})$.

1.4.3.2 MAP estimacija skrivenih tačnih labela

Neka su dati skup podataka D i parametri modela $\phi=\{w,\alpha,\beta\}$. Probabilističke labele su: $z_i = \Pr[y_i = 1|y_i^1, \dots, y_i^R, x_i, \phi]$. Primenom Bajesovog pravila dobijamo:

$$z_i = \frac{\Pr[y_i^1, \dots, y_i^R|y_i = 1, \phi] \cdot \Pr[y_i = 1|x_i, \phi]}{\Pr[y_i^1, \dots, y_i^R|\phi]} \tag{36}$$

Imenilac u formuli (36) možemo dekomponovati na sledeći način:

$$\begin{aligned} &\Pr\left[y_i^1, \dots, y_i^R \middle| y_i = 1, \phi\right] \\ &= \Pr[y_i^1, \dots, y_i^R|y_i = 1, \alpha] \cdot Pr[y_i = 1|x_i, w] \\ &+ \Pr[y_i^1, \dots, y_i^R|y_i = 0, \beta] \cdot \Pr[y_i = 0|x_i, w] \end{aligned} \tag{37}$$

Prema *GMM-MAPML* modelu, *j*-ti anotator koji dobije instancu *xi* za koju treba da odredi klasno obeležje, pronalazi *q*-tu komponentu mešavine koja najverovatnije generiše datu instancu. Nakon toga, anotator generiše labelu sa senzitivnošću α_j^q i specifičnošću β_j^q. Zbog toga važi:

$$\Pr[y_i^1, \dots, y_i^R|y_i = 1, \alpha] = \Pr[y_i^1, \dots, y_i^R|y_i = 1, \alpha_q^1, \dots, \alpha_q^R] \tag{38}$$

gde $q = argmax_{k=1..K}(\tau_{ik})$. Pretpostavka je da svaki od anotatora anotira svaku instancu nezavisno, odnosno da su, pod uslovom da je poznata (tačna) labela y_i, labele $y_i^1, \dots, y_i^R$ nezavisne. Zbog toga važi:

$$\begin{aligned} \Pr\left[y_i^1, \dots, y_i^R \middle| y_i = 1, \alpha_q^1, \dots, \alpha_q^R\right] &= \prod_{j=1}^{R} \Pr\left[y_i^j \middle| y_i = 1, \alpha_q^j\right] = \\ &= \prod_{j=1}^{R} [\alpha_q^j]^{y_i^j} [1 - \alpha_q^j]^{1-y_i^j} \end{aligned} \tag{39}$$

Slično se pokazuje da važi:

$$\Pr[y_i^1, \dots, y_i^R|y_i = 0, \beta] = \prod_{j=1}^{R} [1 - \beta_q^j]^{y_i^j} [\beta_q^j]^{1-y_i^j} \tag{40}$$

Na osnovu (35), (36), (37), (38), (39) i (40) aposteriorna verovatnoća z_i, koja predstavlja probabilističku estimaciju tačne labele se može izračunati na sledeći način:

$$z_i = \frac{a_i p_i}{a_i p_i + b_i(1 - p_i)} \tag{41}$$

gde je:

$$p_i = PR[y_i = 1|x_i, w] = \sigma(w^T x_i)$$

$$a_i = \prod_{j=1}^{R} [\alpha_q^j]^{y_i^j} [1 - \alpha_q^j]^{1-y_i^j}$$

$$b_i = \prod_{j=1}^{R} [1 - \beta_q^j]^{y_i^j} [\beta_q^j]^{1-y_i^j}$$

$$q = argmax_{k=1..K}(\tau_{ik}).$$

1.4.3.3 Model Gausovih mešavina

Model Gausovih mešavina (*Gaussian Mixture Model, GMM*) se najćešće primenjuje u svrhe klasterizacije podataka. Neka je dat skup podataka $D = (x_1, \ldots, x_N)$, gde je svaka komponenta x_i d-dimenzionalni vektor. Pretpostavimo da su opservacije x_i generisane nezavisno iz iste raspodele opisane gustinom verovatnoće $p(x)$ (i.i.d.). *GMM* spada u grupu modela mešavine konačno mnogo raspodela (*finite mixture models*) koji funkciju gustine verovatnoće p(x) modeluju kao težinsku sumu gustina verovatnoće konačnog broja komponenata:

$$p(x|\theta) = \sum_{k=1}^{K} \pi_k f_k(x|z_k, \theta_k), \tag{42}$$

gde:

- K predstavlja broj komponenata;
- $f_k(x|z_k, \theta_k)$ predstavlja k-tu komponentu mešavine. Svaka komponenta mešavine odgovara jednom klasteru i predstavlja gustinu verovatnoće određenu parametrima θ_k);
- $Z = (z_1,...,z_K)$ predstavlja vektor slučajnih promenljivih takvih da uvek samo jedna od promenljivih z_k ima vrednost 1, dok sve ostale imaju vrednost 0. Z reprezentuje identitet komponente koja je generisala opservaciju x;
- π_k predstavlja verovatnoću da je x generisano od strane komponente k ($\sum_{k+1}^{K} \pi_k = 1$). Ovaj model pretpostavlja da je svaka opservacija x_i generisana isključivo od strane jedne od komponenata mešavine.

Ukupan skup parametara navedenog modela je $\theta = \{\pi_1, \ldots, \pi_K, \theta_1, \ldots, \theta_K\}$. Ukoliko su poznati parametri θ, odgovornost τ_{ik} klastera k za opservaciju x_i se može izračunati na sledeći način:

$$\tau_{ik} = p(z_{ik} = 1|x_i, \theta) = \frac{\pi_k f_k(x_i|z_k, \theta_k)}{\sum_{m=1}^{K} \pi_m f_m(x_i|z_m, \theta_m)}. \tag{43}$$

Odgovorornost τ_{ik} reflektuje nesigurnost (*uncertanty*) koja od K komponenti je generisala opservaciju x_i. Na ovaj način se problem svodi na estimaciju parametara pretpostavljenog modela mešavina, odnosno na estimaciju stvarne gustine verovatnoće podataka na osnovu opservacija.

GMM uvodi pretpostavku da svaka komponenta predstavlja multivarijantnu Gausovu distribuciju. Komponenta k je opisana parametrima $\theta_k = \{\mu_k , \Sigma_k\}$ i gustinom verovatnoće:

$$f_k(x_i|\mu_k, \Sigma_k) = \frac{exp\left\{-\frac{1}{2}(x_i-\mu_k)^T \Sigma_k^{-1}(x_i-\mu_k)\right\}}{(2\pi)^{d/2}|\Sigma_k|^{1/2}}, \quad (44)$$

gde μ_k predstavlja d-dimenzioni vektor srednjih vrednosti, Σ_k predstavlja $d \times d$ kovarijansnu matricu. Kovarijansna matrica Σ_k određuje geometrijske osobine klastera određenog komponentom k.

Obučavanje GMM modela se svodi na estimaciju parametara $\theta = \pi_1, ..., \pi_K, \mu_1, ..., \mu_K, ..., \Sigma_1, ..., \Sigma_K$. Logaritamska verodostojnost mešavine je:

$$L(\theta|x_1, \dots, x_N) = \sum_{i=1}^{N} ln\{\sum_{k=1}^{K} \pi_k f_k(x_i|\mu_k, \Sigma_k)\}) \quad (45)$$

Glavna prepreka prilikom estimacije parametara *GMM* modela jeste što nije poznato koja latentna komponenta je odgovorna za generisanje kojih opservacija. U slučaju kada je poznato koje opservacije pripadaju kojoj komponenti, veoma je jednostavno odrediti parametre Gausove raspodele za svaki od skupova opservacija.

Najčešće primenjivana tehnika za estimaciju parametara kod modela mešavine konačnog broja raspodela jeste *EM* metoda. Međutim, da bi se primenio ovaj algoritam neophodno je specificirati broj komponenata u mešavini K, kao i formu gustina verovatnoće komponenata[6]. Različite kombinacije broja klastera i formi gustina verovatnoća komponenti možemo posmatrati kao različite statističke modele podataka. Određivanje finalnog modela koji najbolje odgovara podacima je moguće izvesti primenom Bajesovog informacionog kriterijuma (*Bayesian Information Criterion, BIC*).

1.4.3.3.1 *EM* metoda za estimaciju parametara *GMM* modela

EM je iterativni algoritam koji počinje inicijalnom estimacijom parametara modela θ i zatim iterativno ažurira θ sve dok se ne ispuni neki zadati uslov konvergencije. Algoritam 2 prikazuje *EM* metodu za estimaciju parametara *GMM* modela.

[6] U slučaju *GMM* modela ovo se svodi na formu kovarijansnih matrica Σ_k.

Ulaz
Opservacije $D = (x_1, \ldots, x_N)$, broj Gausovih komponenti K i forma kovarijansne matrice
Izlaz
Parametri modela $\theta = \pi_1, \ldots, \pi_K, \mu_1, \ldots, \mu_K, \ldots, \Sigma_1, \ldots, \Sigma_K$ i odgovornosti τ_{ik} komponente k za opservaciju x_i, i=1,…,N; k= 1,…, K.
EM postupak
1. Primenom algoritma *k*-sredina (*k-means*) inicijalizovati vektor sredina μ_k, kovarijanse Σ_k i koeficijente mešavine π_k i izračunati logaritamsku verodostojnost primenom formule (45). 2. (*E*-korak) Korišćenjem trenutnih vrednosti parametara modela izračunati odgovornosti τ_{ik} primenom formule (43). 3. (*M*-korak) Koristeći trenutnu estimaciju odgovornosti τ_{ik} ponovo estimirati parametre modela. Neka je $N_k = \sum_{i=1}^{N} \tau_{ik}$, odnosno suma odgovornosti *k*-te komponente – ovo je efektivan broj opservacija dodeljen komponenti *k*, te se prema tome težine mešavina mogu ažurirati na sledeći način: $$\pi_k^{new} = \frac{N_k}{N} \quad (46)$$ Sada se μ_k može ažurirati pronalaženjem srednje vrednosti uz upotrebu odgovornosti: $$\mu_k^{new} = \frac{1}{N_k} \sum_{i=1}^{N} \tau_{ik} x_i \quad (47)$$ Kovarijansu možemo izračunati kao: $$\Sigma_k^{new} = \frac{1}{N_k} \sum_{i=1}^{N} \tau_{ik} (x_i - \mu_k^{new})(x_i - \mu_k^{new})^T \quad (48)$$ 4. Korišćenjem ažuriranih vrednosti prarametara, na osnovu formule (45) izračunati logaritamsku verodostojnost. Proveriti da li je ispunjen kriterijum konvergencije (propisan za parametre ili za logaritamsku verodostojnost). Ukoloko kriterijum konvergencije nije ispunjen, vratiti se na korak 2.

Algoritam 2: estimacija parametrara *GMM* modela pomoću *EM* metode

1.4.3.3.2 **Bajesova selekcija modela**

Svaka kombinacija broja klastera i forme gustina verovatnoća komponenti predstavlja određeni statistički model podataka. Jedna od prednosti *GMM* jeste što nam omogućava korišćenje približnih Bajesovih faktora za poređenje modela. Ovo je sistematski način istovremene selekcije parametrizacije modela kojim se predstavljaju podaci i broja komponenti.

Neka je dat skup opservacija X i neka su M_1 i M_2 dva modela sa parametrima θ_1 i θ_2, respektivno. Integral verodostojnosti (verovatnoća uočavanja opservacija X pod uslovom da se u osnovi nalazi model M_g, g=1,2) je:

$$p(X|M_g) = \int p(X|\theta_g, M_g)p(\theta_g|M_g)d\theta_g. \quad (49)$$

Bajesov faktor se definiše kao odnos integrala verodostojnosti dva modela:

$$B_{1,2} = P(X|M_1)/P(X|M_2). \quad (50)$$

Vrednost faktora $B_{1,2}$ veća od 1 označava da je verovatnije da su podaci distribuirani prema modelu M_1 nego prema modelu M_2. Moguća je i generalizacija poređenja modela na više od dva modela. Najveća poteškoća na koju nailazimo prilikom primene Bajesovog faktora jeste izračunavanje integrala verodostojnosti. Međutim, moguće je primeniti aproksimaciju koja se zove Bajesov informacioni kriterijum (*Bayesian Information Crtiterium, BIC*) [Fraley 1998]:

$$p(X|M_g) \approx BIC_g = 2log(X|\widehat{\theta_g}, M_g) - m_g \log(N), \quad (51)$$

gde je m_g broj nezavisnih parametara modela M_g, a $\widehat{\theta_g}$ je estimacija parametra θ_g za koju je verodostojnost maksimalna. Velika *BIC* vrednost ukazuje da postoji snažan dokaz za postojanje odgovarajućeg modela. Stoga je moguće koristiti *BIC* vrednost za poređenje modela koji imaju različite parametrizacije kovarijansnih matrica i različit broj komponenti – model sa najvećom *BIC* vrednošću se bira kao 'najbolji' model. Algoritam 3 prikazuje proceduru selekcije 'najboljeg' modela Gausovih mešavina.

Ulaz
Skup opservacija $X = (x_1, \ldots, x_N)$
Izlaz
Optimalan broj komponenti, optimalna forma komponenti *GMM* modela i odgovarajući parametri modela i odgovornosti τ_{ik} komponente k za opservaciju x_i, i=1,…,N; k= 1,…, K.
Algoritam
1. Odabrati formu modela *M* iz 9 kandidatskih modela [Martinez 2004]. 2. Odabrati broj komponenti *K* (minimalan broj komponenti je 1, a maksimalan 6). 3. Primenom algoritma 2 pronaći parametre modela određenog sa *M* i *K* i njegovu logaritamsku verodostojnost. 4. Primenom jednačine (50) izračunati *BIC* vrednost modela odreženog sa *M* i *K*. 5. Vratiti se na korak 2 i odabrati novu vrednost za *K*. 6. Vratiti se na korak 1 i odabrati novu formu modela *M*. 7. Odabrati optimalnu konfiguraciju (broj komponenti i forma kovarijansnih matrica) koja odgovara najvećoj dobijenoj *BIC* vrednosti.

Algoritam 3: Bajesova selekcija modela za *GMM*

1.4.3.4 Sumarizacija *GMM-MAPML* algoritma

Sumarizacija *GMM-MAPML* algoritma je predstavljena algoritmom 4. Detalji vezani za *GMM-MAPML* algoritam se mogu naći u [Zhang 2011].

Ulaz
Obučavajući skup $D = \{x_i, y_i^1, \ldots, y_i^R\}$, $i = 1..n$ gde x_i predstavlja i-tu instancu skupa D, a $y_i^j \in \{0,1\}$ predstavlja binarno klasno obeležje instance x_i, dodeljeno od strane j-og anotatora
Izlaz
Skup $T = \{(x_i, y_i)\}$, $i = 1..n$ gde x_i predstavlja i-tu instancu ulaznog obučavajućeg skupa D, a y_i estimaciju tačnog klasnog obeležja instance x_i.
Algoritam
1. Primenom algoritma 3 estimirati parametre *GMM* modela sa K komponenti koji najviše odgovara distribuciji instanci obučavajućeg skupa, kao i odgovornosti τ_{ik} za svaku instancu x_i, $i = 1..n$, $k = 1..K$. 2. Inicijalizovati za svaku instancu x_i, $i = 1..n$, inicijalizovati estimiranu labelu z_i pomoću većinskog glasanja: $z_i = (1/R)\sum_{j=1}^{R} y_i^j$. 3. Iterativna optimizacija a. (ML estimacija) Za datu labelu z_i, estimirati senzitivnost i specifičnost j-tog anotatora za k-tu komponentu primenom formula (33) i (34). Takođe, obučiti model logističke regresije uz pomoć *Newton-Raphson* metode radi optimizacije parametra w. b. (MAP estimacija) Na osnovu parametara modela $\phi = \{w, \alpha, \beta\}$, ažurirati z_i primenom (41). 4. Ukoliko ϕ i z ostanu nepromenjeni tokom dve sukcesivne iteracije ili je izvršen maksimalan broj iteracija, preći na korak 5., inače se vratiti na korak 3. 5. Estimirati skrivenu tačnu labelu y_i primenom praga γ na z_i, t.j. $y_i = 1$ ukoliko važi da je $z_i > \gamma$, inače $y_i = 0$.

Algoritam 4: Sumarizacija *GMM-MAPML* algoritma

GMM-MAPML algoritam je moguće proširiti radi primene na skupove podataka sa više kategorija. Pretpostavimo da je instance skupa podataka neophodno razvrstati u $C \geq 2$ klase. Neka su $y_i^j \in \{1, \ldots, C\}$ labele dodeljene i-toj instanci od strane j-tog anotatora i neka $y_i \in \{1, \ldots, C\}$ prestavljaja tačnu labelu i-te instance. Svakog anotatora za k-tu komponentu modelujemo pomoću multinominalnih parametara $\alpha_t^{j(k)} = (\alpha_{t1}^{j(k)}, \ldots, \alpha_{tC}^{j(k)})$, gde

$$\alpha_{tc}^j = \Pr[y_i^j = c | y_i = t], \sum_{c=1}^{C} \alpha_{tc}^j = 1.$$

Paratmetar α_{tc}^{j} označava verovatnoću da je anotator *j* dodelio klasu *c* instanci ukoliko je prava klasa instance *t*. Na ovaj način se algoritam 4 može primeniti na višekategorijske skupove podataka.

2 Pregled vladajućih stavova i shvatanja u literaturi

U ovom poglavlju biće dat detaljan pregled postojećih pristupa primene ko-trening algoritma na skupove podataka bez prirodne podele obeležja. U postojećoj literaturi postoje dva generalna pravca istraživanja mogućnosti primene ko-treninga na skupove podataka bez prirodne podele obeležja. Prvi pravac predstavlja istraživanje sa ciljem razvoja metodolgije za pronalaženje optimalne veštačke podele obeležja koja bi garantovala visoke performanse ko-trening algoritma. Drugi pravac predstavljaju strategije kombinovanja ko-treninga sa tehnikama učenja sa grupom hipoteza sa ciljem mogućnosti primene ko-treninga na skupove podataka bez prirodne podele obeležja kao i pobojšanja performansi ko-trening algoritma.

2.1.1 Ko-trening primenjen sa veštačkom podelom obeležja

Inicijalne studije ko-trening algoritma su se fokusirale na mogućnost njegove primene i razjašnjavanja uslova pod kojima se ko-trening može efektivno primeniti u cilju povećanja performansi inicijalnog slabog klasifikatora treniranog na malom skupu anotiranih podataka. U ovom cilju kreirane su raznovrsne metodologije za kreiranje veštačke podele obeležja i ispitivane zavisnosti performansi ko-treninga od karakteristika dobijenih pogleda. U ovom odeljku biće izložen pregled rezultata i zaključaka ovih studija.

2.1.1.1 Slučajna podela obeležja

Pod slučajnom podela obeležja se u literaturi podrazumeva da se svako od obeležja skupa podataka na slučajan način dodeli jednom od dva rezultujuća pogleda. Pogledi se međusobno ne preklapaju i sadrže približno jednak broj obeležja.

U [Nigam 2000b] je izvedena empirijska studija osetljivosti performansi ko-trening algoritma na karakteristike korišćenih pogleda. Njihovi eksperimenti su pokazali da su performanse ko-treninga zaista umanjene narušavanjem pretpostavke o uzajamnoj nezavisnosti pogleda, ali ne u meri u kojoj je očekivano. Zaključak ove studije studije je da u slučaju postojanja adekvatne prirodne podele obeležja ko-trening algoritam rezultuje većim performansama od polu-nadgledanih tehnika koje datu podelu ne koriste. Takođe, ukoliko prirodna podela obeležja ne postoji, a skup podataka se ogleda dovoljnom redundantnošću, ko-trening algoritam primenjen sa slučajnom podelom obeležja može rezultovati većim performansama u odnosu na druge tehnike polu-nadgledanog obučavanja. Pod redudantnošću se u ovom kontekstu podrazumeva da je moguće obučiti klasifikator relativno velike tačnosti korišćenjem svakog

od pogleda zasebno, odnosno, da se tačnost klasifikatora ne bi u značajnoj meri smanjila ukoliko bi smo koristili isključivo obeležja jednog od pogleda [Chan 2004; Feger 2006].

Efektivnost korišćenja slučajne podele obeležja radi primene ko-trening algoritma potvrđena je i u kasnijim empirijskim studijama [Chan 2004; Koprinska 2007]. U eksperimentima sprovedenim u [Nigam 2000b] prirodna podela se pokazala efektivnijom od slučajne podele obeležja. Međutim, [Chan 2004] pokazuju da performanse ko-treninga primenjenog sa slučajnom podelom obeležja mogu biti uporedive, ili čak i prevazići performanse ko-treninga sa prirodnom podelom obeležja u slučaju velike razlike u kvalitetu (u smislu tačnosti klasifikacije) klasifikatora nastalih korišćenjem prirodne podele obeležja. Ovi rezultati nagoveštavaju da ko-trening može biti prilično robustan na pretpostavku o uslovnoj nezavisnosti pogleda, sve dok postoji dovoljna redudantnost u skupu obeležja. To bi značilo da je moguće pobojšati performanse ko-treninga pronalaženjem ravnoteže između redudantnosti pogleda i njihove međusobne nezavisnosti.

Iako se slučajna podela obeležja u proseku odlikovala visokim performansama ko-treninga, ovo nije optimalna postavka jer ne postoji garancija da će pojedinačna, konkretna slučajna podela rezultovati unapređenjem performansi polaznog klasifikatora. Naime, slučajna podela obeležja često rezultuje pogledima koji nisu međusobno uslovno nezavisni. Zbog ovoga rezultujući klasifikatori mogu da se ponašaju slično u smislu da isti primeri mogu biti pogrešno anotirani od strane oba klasifikatora. U ekstremnom slučaju gde su svi pogrešno anotirani primeri isti za oba klasifikatora, ko-trening proces se svodi na samo-obučavanje. Štaviše, budući da je za treniranje pojedinačnih klasifikatora korišćen manji skup obeležja nego što bi bio slučaj u samo-obučavanju, performanse ko-treninga sa slučajnom podelom u ovom slučaju mogu biti manje nego performanse dobijene samo-obučavanjem.

Nemogućnost garancije performansi čini ovaj model neprimenljivim u praksi. Jedan od važnih zaključaka studije [Nigam 2000b] jeste da su performanse ko-treninga osetljive na ispunjenost uslova nezavisnosti pogleda, što je motivisalo mnoge autore da uključe ovaj uslov u dizajn optimalne veštačke podele.

2.1.1.2 Podela obeležja bazirana na uslovu maksimalne nezavisnosti pogleda

Jedan od uslova pod kojim je uspešna primena ko-treninga zagarantovana jeste da su pogledi uslovno nezavisni ukoliko je poznato klasno obeležje [Blum 1998]. Na osnovu ovog uslova razvijena je *maxInd* metodologija za kreiranje veštačke podele obeležja, dizajnirana da maksimizuje uslovnu nezavisnost pogleda u odnosu na klasno obeležje [Feger 2006]. Kao mera uslovne zavisnosti pogleda u odnosu na klasu korišćena je uslovna uzajamna

informativnost (*Conditional Mutual Information*, *CondMI*) koja predstavlja deljenu količinu informacije između dva obeležja pod uslovom da je klasno obeležje poznato [MacKay 2003]. Mana ovog pristupa je što *CondMI* mera zahteva poznavanje klasnog obeležja primera, zbog čega je podelu moguće izvesti isključivo na osnovu malog skupa anotiranih podataka, što ne mora da rezultuje optimalnim rešenjem.

Eksperimentalni rezultati sa *maxInd* metodologijom su pokazali da, u slučaju ispunjenosti prvog uslova o dovoljnoj redudantnosti pogleda, korišćenje maksimalno nezavisnih pogleda ne rezultuje boljim performansama od korišćenja slučajne podele. Kao moguće objašnjenje ovog fenomena autori navode uticaj međusobne zavisnosti obeležja istog pogleda na ko-trening – minimizacija nezavisnoti među pogledima dovodi do maksimizacije zavisnoti obeležja unutar svakog od pogleda. Autori zaključuju da bi bolje rešenje predstavljao kompromis između zavisnosti obeležja unutar svakog od pogleda i zavisnosti između samih pogleda i ukazuju da je neophodno preispitati postavljeni uslov o nezavisnosti pogleda za uspešnu primenu ko-treninga. Kasnije su ovi rezultati potvrđeni i u [Terabe 2008], gde je generisan veći broj slučajnih podela, nakon čega su poređene performanse ko-treninga primenjenog na podele veće međusobne zavisnosti u odnosu na performanse ko-treninga primenjneog na podele manje međusobne zavisnosti. U eksperimentima se pokazalo da ko-trening primenjen sa veštačkom podelom obeležja ponekad ne rezultuje dobrim performansama, čak i ako podela zadovoljava uslove postavljene u [Blum 1998], što znači da je neophodno dalje istraživanje zahteva koje bi podela obeležja trebala da ispuni.

2.1.1.3 Mešoviti kriterijumi podele

Eksperimenti pokazuju da istovremena optimizacija oba kriterijuma koja podela obeležja treba da ispuni zarad uspešne primene ko-treninga daje bolje rezultate od optimizacije samo jednog od ovih kriterijuma [Salaheldin 2010]. Prvi uslov uspešne primene ko-treninga jeste redudantnost pogleda, odnosno mogućnost da se na osnovu individualnih pogleda mogu obučiti klasifikatori dovoljne tačnosti. Budući da je zbog nedostatka anotiranih primera nemoguće direktno meriti tačnost obučenih klasifikatora, autori u [Salaheldin 2010] kao meru pouzdanosti individualnih pogleda predlažu entropiju njihovih izlaza. U cilju kreiranja podele koja će rezultovati najpouzdanijim pogledima, minimizuje se suma entropija izlaza individualnih klasifikatora. Drugi uslov uspešne primene ko-treninga jeste međusobna nezavisnost pogleda. Kao meru zavisnosti pogleda autori u [Salaheldin 2010] su usvojili su *CondMI* meru predloženu u [Feger 2006]. Kao mešoviti kriterijum podele autori minimizuju zbir navedenih mera.

Konačno, [Salaheldin 2010] predlažu još jedan kriterijum podele koji teži da maksimizuje jačinu individualnih pogleda, kao i njihovu međusobnu

različitost. Nezavisnost pogleda je teorijski važan faktor uspešnosti ko-trening algoritma. Međutim, rezultati u [Feger 2006] pokazuju da prosta optimizacija ovog kriterujma često ne rezultuje boljim performansama od slučajne podele. [Salaheldin 2010] navode da je razlog tome što se *maxInd* algoritam može primeniti samo na malom skupu anotiranih primera kojima raspolažemo u ko-trening postavci (budući da mu je za optimizaciju neophodo poznavanje klasnog obeležja), zbog čega proizvedeni pogledi nisu bili u značajnoj meri bolji po pitanju međusobne nezavisnosti od pogleda dobijenih slučajnom podelom obeležja. Zbog toga, [Salaheldin 2010] kao meru različitosti pogleda predlažu razliku pouzdanosti individualnih klasifikatora iz grupe klasifikatora i njihovog kombinovanog izlaza. Ovo reprezentuje pobojšanje kombinovanog klasifikatora u odnosu na individualne klasifikatore. Ovakav kriterijum podele je u njihovim eksperimentima pokazao performanse slične mešovitom kriterijumu podele.

U radu [Du 2010] predložen je metod empirijske evaluacije ispunjenosti dve pretpostavke za uspešnu primenu ko-treninga. U ovom radu su takođe predstavljena tri nova pristupa za podelu obeležja skupova podataka kod kojih prirodna podela obeležja ne postoji ili nije poznata. Prvi metod je baziran na meri entropije atributa, sličnoj onoj koja se koristi kod stabala odlučivanja [Quinlan 1993]. Intuitivno, što je entropija atributa veća, veća je i njegova prediktivna moć. U ovom pristupu se atributi najveće entropije ravnomerno distribuiraju između dva pojedinačna pogleda. Ovaj metod teži da maksimizuje prediktivne performanse pojedinačnih klasifikatora (bazirano pretpostavci da svaki klasifikator ponaosob mora da bude dovoljan za klasifikaciju), ali ne uzima u obzir uslov o međusobnoj nezavisnosti pogleda. Preostala dva pristupa uzimaju u obzir i drugi kriterijum radi povećanja performansi ko-treninga i pokazuju superiorne performanse u odnosu na prvi kriterijum. Eksperimentalno je pokazano da su predložene tehnike za evaluaciju i podelu skupa obeležja veoma uspešne ukoliko raspolažemo dovoljno velikim skupom podataka. Međutim, u praktičnim primenama gde bi ko-trening bio od najveće koristi raspolažemo sa veoma malim skupovima podataka. Eksperimenti izvedeni u [Du 2010] na realnim, ali i sintetičkim podacima, generisanim specifično za potrebe ko-treninga, pokazuju da ispunjenost preduslova uspešne primene ko-treninga ne može biti pouzdano verifikovana na malim skupovima podataka, zbog čega metode za podelu obeležja nisu pouzdane. U datim situacijama ko-trening nije pozdan.

2.1.1.4 Zaključak

U ovom odeljku su izloženi mnogi obećavajući rezultati u pogledu pronalaženja optimalne veštačke podele obeležja pomoću koje bi se ko-trening mogao primeniti na skupove podataka bez prirodne podele obeležja. Međutim, nijedna od razvijenih metodologija ne garantuje dobre performanse ko-treninga na svakom skupu podataka na koji je primenjena. Dizajn univerzalne metodologije za kreiranje idealne veštačke podele se pokazao kao težak zadatak.

Eksperimenti pokazuju da zavisnost karakteristika pogleda i performansi ko-treninga nije u dovoljnoj meri ispitana [Feger 2006; Terabe 2008], kao i da je u ko-trening postavci, gde raspolažemo samo malim brojem anotiranih primera, često i nemoguće evaluirati karakteristike konstruisanih pogleda [Du 2010].

Pri svemu tome, performanse ko-trening algoritma su veoma zavisne od korišćene podele obeležja i ostalih ulaznih parametara [Nigam 2000b; Pierce 2001]. Ukoliko nisu ispunjeni uslovi za njegovu uspešnu primenu, ko-trening može da rezultuje i degradacijom performansi u odnosu na inicijalni slabi klasifikator time što unosi šum u inicijalni skup podataka [Pierce 2001].

Zbog svega navedenog, dizajn metodologije za pronalaženje idealne veštačke podele osobina ostaje otvoren problem.

2.1.2 Kombinovanje ko-treninga sa tehnikama učenja sa grupom hipoteza

Drugi pravac istraživanja mogućnosti primene ko-treninga na skupove podataka bez prirodne podele jeste kombinovanje ko-trening algoritma sa tehnikama učenja sa grupom hipoteza.

U [Zhou 2009] je teorijski pokazano da kombinovanje različitih klasifikatora može biti veoma korisno u polu-nadgledanoj postavci baziranoj na neslaganju (*disagreement-based*), kakva je i ko-trening postavka. Navode se dva razloga: 1) sa izvršavanjem ko-trening procesa klasifikatori postaju međusobno sve sličniji, tako da performanse individualnih klasifikatora stagniraju posle određenog broja iteracija. Međutim, eksploatacijom grupe klasifikatora moguće je dalje smanjiti grešku generalizacije pomoću samo-obučavanja; 2) kombinacija klasifikatora može dostići dobre performanse brže (u manje iteracija) od individualnih klasifikatora.

Ko-trening metod baziran na tri unutrašnja klasifikatora koja su inicijalizovana na različitim podskupovima inicijalnog anotiranog skupa (kao u *bagging* proceduri) je predstavljen u [Zhou 2005a]. Ovi klasifikatori se zatim iterativno unapređuju – u svakoj iteraciji ko-treninga se instance anotiraju i dodaju u obučavajući skup pojedinačnog klasifikatora ukoliko se preostala dva klasifikatora slažu u pogledu njihovih anotacija. Na ovaj način se izbegava potreba za ekspicitnim merenjem pouzdanosti anotacije instance od pojedinačnog klasifikatora. Predikcije pojedinačnih klasifikatora se na kraju algoritma kombinuju primenom većinskog glasanja. Nedostaci ovog pristupa su što je broj klasifikatora grupe prema svom dizajnu ograničen na tri, dok bi se sa povećanjem veličine grupe očekivalo i pobojšanje u performansama. Takođe, *bagging* tehnika se primenjuje samo u inicijalnoj iteraciji i nije proučavano kako ko-trening porces utiče na raznolikost kreirane grupe klasifikatora – moguće je da opisan način dodavanja primera u obučavajući skup rezultuje time da

pojedinačni klasifikatori postaju međusobno sve sličniji, što narušava performanse formirane grupe klasifikatora.

Ovaj metod proširen je u [Li 2007] u postavci nazvanoj *Co-Forest*. U ovom proširenju se koristi $N \geq 3$ klasifikatora u grupi, označenoj sa H^*. Grupa klasifikatora je korišćena u svrhu efikasnije estimacije pouzdanosti anotiranih instanci – prilikom određivanja najpouzdanije anotiranih instanci koje se dodaju u obučavajući skup pojedinačnog klasifikatora h_i ($i = 1,\ldots,N$), koriste se svi klasifikatori grupe H^* osim samog klasifikatora h_i. Ovako formirana grupa klasifikatora se naziva prateći ansambl (*concomitant ensemble*) i označava sa H_i ($= H^* - h_i$). Pouzdanost anotacije instance se estimira stepenom slaganja anotacija dodeljenih od strane grupe H_i. *Co-Forest* prvo konstruiše inicijalnu grupu klasifikatora primenom metode slučajne šume (*random forest*). Zatim se u svakoj iteraciji *Co-Forest* algoritma u obučavajući skup klasifikatora h_i dodaju instance koje su pouzdano anotirane od strane njegovog pratećeg ansambla H_i. Ovi primeri se ne uklanjaju iz skupa neanotiranih primera, kako bi ponovo mogli biti selektovani od strane nekog drugog pratećeg ansambla H_j u narednim iteracijama. Kao rezultat ovog procesa, obučavajući skupovi različitih klasifikatora postaju sve sličniji. Ukoliko bi *Co-Forest* bio baziran na tehnici učenja sa grupom hipoteza koja različitost u grupi klasifikatora postiže korišćenjem različitih obučavajućih skupova pri obučavanju klasifikatora (kao što je slučaj sa *bagging* tehnikom), opisani iterativni proces *Co-Forest* algoritma bi smanjio različitost klasifikatora u grupi, što se odražava smanjenjem tačnosti dobijene kombinacijom grupe. Zbog toga se u cilju održanja različitosti klasifikatora u grupi *Co-Forest* zasniva na tehnici slučajne šume, zasnivanoj na manipulaciji skupa obeležja umesto na manipulaciji obučavajućeg skupa.

Glavni nedostatak *Co-Forest* algoritma jesu ograničenja koja postavlja u pogledu tehnike obuke grupe klasifikatora, kao i broja klasifikatora u grupi koji ne može da bude previše velik. Naime, eksperimenti u [Li 2007] pokazuju da se najbolje performanse postižu ukoliko broj klasifikatora nije prevelik (4-6 klasifikatora). U slučaju velikog broja klasifikatora (100) pobojšanje postaje veoma malo i čak dolazi do degradacije performansi u odnosu na polazni klasifikator. Ovo se dešava zbog toga što, u slučaju većeg broja klasifikatora, *Co-Forest* svojim obučavajućim procesom narušava međusobnu raznolikost grupe klasifikatora: dva prateća ansambla H_i i H_j (zadužena za selekciju pouzdanih primera koji će biti dodati u obučavajuće skupove klasifikatora h_i, odnosno h_j) se međusobno razlikuju samo u pogledu dva klasifikatora – h_i i h_j. Ukoliko je formirana grupa klasifikatora ansambla velika, H_i i H_j će biti međusobno veoma slični, što rezultuje dodavanjem istih primera u obučavajuće skupove klasifikatora h_i i h_j. Ovo rezultuje time da klasifikatori h_i i h_j u toku obučavajućeg procesa postaju sve sličniji. Drastično opadanje različitosti klasifikatora sa procesom obučavanja drastično narušava performanse cele grupe. Eksperimenti [Li 2007] potvrđuju ovu tvrdnju – ovaj efekat je najizraženiji u slučaju manjeg broja anotiranih primera u ondosu na neanotirane.

U [Hady 2008a] predstavljen je okvir (*framework*) za polu-nadgledano obučavanje nazvan ko-trening sa komitetom (*Co-Training with Committee, CoBC*) koji predstavlja proširenje *Co-Forest* pristupa. Kao i kod *Co*-Forest pristupa, inicijalno se izgrađuje komitet od *N* međusobno različitih klasifikatora relativno dobrih performansi. Zatim se, iterativno, grupa klasifikatora primenjuje na skup neanotiranih podataka u cilju selekcije najpouzdanije anotiranih primera i njihove anotacije. Ovako anotirani primeri se dodaju u obučavajući skup, nakon čega se prethodni komitet klasifikatora odbacuje i formira se novi na osnovu novog (uvećanog) obučavajućeg skupa. Formiranje novog komiteta omogućava održavanje raznolikosti među njegovim članovima. Opisano okruženje je moguće primeniti u kombinaciji sa bilo kojom tehnikom učenja sa grupom hipoteza, a u eksperimentima u [Hady 2008a] su ispitivane *bagging* tehnika [Breiman 1996], *AdaBoost* [Freund 1996] tehnika i metoda slučajnih potprostora (*random subspace method, RSM*). U izvršenim eksperimentima *CoBC* varijante su pokazale bolje performanse od *bagging*, *AdaBoost* i *RSM* tehnike u slučaju kada je broj anotiranih podataka ograničen. Takođe, *CoBC* je pokazao bolje performanse i od samo-obučavanja, kao i od ko-treninga primenjenog sa slučajnom podelom obeležja.

U [Freund 1997] ko-trening je kombinovan sa tehnikama učenja sa grupom hipoteza u postavci za aktivno učenje. U ovom pristupu iterativno se konstruiše grupa klasifikatora (komitet) bazirana na trenutnom obučavajućem skupu. Svaki od članova formiranog komiteta glasa o labeli svake neanotirane instance i za svaku instancu se meri neslaganje članova komiteta. Ovim postupkom se određuje koje su instance najinformativnije i takve instance se prosleđuju ljudskom anotatoru koji im dodeljuje tačno klasno obeležje.

U opisanim pristupima omogućena je primena ko-treninga na skupove podataka za koje nije poznata prirodna podela obeležja na taj način što se dva pojedinačna klasifikatora, definisana u originalnom ko-trening algoritmu, zamenjuju grupom klasifikatora. Ovi pristupi su pokazali veliki uspeh u podizanju performansi ko-trening algoritma. Glavni nedostatak opisanih algoritama jeste potreba za relativno velikim anotiranim skupom (u odnosu na originalnu ko-trening postavku) zbog inicijalizacije grupe klasifikatora. Naime, najmanja veličina inicijalnog anotiranog skupa korišćena u eksperimentima u ovim radovima je 20% skupa podataka, a pokazano je da, ukoliko postoji odgovarajuća podela obeležja, ko-trening proces može da uči polazeći od svega jednog anotiranog primera [Zhou 2007b].

2.1.3 Drugi pristupi

U radu [Goldman 2000] je predstavljen algoritam čiji se način kreiranja dva različita klasifikatora u okviru ko-treninga ne bazira na primeni istog obučavajućeg algoritma na različitim skupima obeležja, već na primeni različitih algoritama obučavanja nad istim skupom obeležja, pod pretpostavkom da svaki od klasifikatora deli ulazni prostor u skup klasa ekvivalencije. Npr. stablo

odlučivanja particioniše ulazni prostor na jednu klasu ekvivalencije za svaki od listova. U ovom algoritmu se najpouzdanije anotirane instance skupa *U* identifikuju primenom desetostruke unakrsne evaluacije i skupa statističkih testova, nakon čega se dodaju u obučavajući skup radi obuke drugog klasifikatora. Desetostruka unakrsna evaluacija se koristi i prilikom kombinovanja hipoteza u cilju dobijanja konačne predikcije. Ovaj pristup je kritikovan da ima sledeće mane: pretpostavke koje ovaj pristup uvodi ograničavaju njegovu mogućnost primene; u cilju određivanja kada bi jedan klasifikator trebao da anotira instancu za drugi korišćeni su statistički testovi, međutim količina anotiranih podataka nije bila dovoljna za aplikaciju ovih testova; konačno, korišćena je desetostruka validacija, a količina raspoloživih anotiranih podataka je takođe previše mala za ovaj postupak, unakrsna validacija je vremenski zahtevna procedura [Hady 2008a].

Kasnije je u [Zhou 2004] je ovaj pristup iz [Hady 2008a] unapređen. Na mestu standardna dva klasifikatora formirana je grupa klasifikatora. Klasifikatori se formiraju korišćenjem skupa različitih obučavajućih algoritama nad istim anotiranim skupom, budući da je zbog ograničenog broja anotiranih primera nemoguće formirati različite obučavajuće podskupove na osnovu anotiranog skupa. Instance neanotiranog skupa *U* su u svakoj iteraciji anotirane primenom težinskog glasanja. Anotirani primeri se u svakoj iteraciji dodaju isključivo u obučavajuće skupove onih klasifikatora koji se nisu slagali sa dodeljenom anotacijom. Ovaj proces se ponavlja sve dok ne nestane primera koji se na opisan način mogu dodati u obučavajuće skupove. Autori takođe kombinuju svoj pristup sa aktivnim učenjem radi postizanja boljih performansi. Ovaj pristup redukuje potrebu za statističkim testovima, ali takođe koristi vremenski zahtevnu unakrsnu validaciju u cilju određivanja intervala pouzdanosti koji se koriste za određivanje najpouzdanije anotiranih instanci i kombinovanje hipoteza radi donošenja konačne odluke [Hady 2008].

U [Wang 2007] je predstavljena teorijska analiza koja razotkriva da je kod pristupa baziranih na neslaganju (*disagreement-based approaches*), kao što je ko-trening, ključ uspeha postojanje velike različitosti među klasifikatorima i da je nevažno iz kog izvora je različitost proistekla. Ovo daje teorijsku podlogu za uspeh ko-trening metoda primenljivih na skupove podataka sa jednim pogledom. Razlog za uspeh svih opisanih ko-trening tehnika jeste kreiranje međusobene različitosti klasifikatora primenom različitih tehnika: korišćenje različitih skupova obeležja nad istim obučavajućim skupom [Blum 1998], različitih obučavajućih algoritama na istom obučavajućem skupu [Goldman 2000; Zhou 2004], manipulacija obučavajućeg skupa primenom *bagging* tehnike [Zhou 2005a] ili manipulacija skupa obeležja primenom tehnike slučajne šume [Li 2007].

3 Metodologija

U ovom poglavlju biće izložen formalan opis metodologije modela predstavljenih u ovoj disertaciji. U disertaciji su predstavljena dva modela: Algoritam statistike slučajnih podela (*Random Split Statistic algorithm, RSSalg*) i Integracija višestrukih ko-trening klasifikatora (*Integration of Multiple Co-trainined Classifiers, IMCC*). Rezultati primene *RSSalg* i *IMCC* modela publikovani su u radovima [Slivka 2010, 2013a i 2013b] i [Slivka 2012b], respektivno. Oba modela u osnovi imaju isti postupak kreiranja grupe nezavisnih ko-trening klasifikatora koji će biti izložen u prvom odeljku ovog poglavlja. Detalji metodologije karakteristični za *RSSalg*, odnosno, *IMCC* model, opisani su u drugom, odnosno trećem odeljku ovog poglavlja, respektivno. Četvrti odeljak ovog poglavlja opisuje motivaciju koja stoji iza definisane metodologije predloženih rešenja.

Neka je dat problem klasifikacije gde je neophodno razvrstati instance zadatog skupa $T = \{t_i\}$ u jednu od K predefinisanih klasa C_k, $k \in \{1..K\}$. Neka pri tome raspolažemo sa malim skupom anotiranih primera $L = \{(l_i, y_i)\}$, gde l_i predstavlja instancu skupa L, a $y_i \in \{C_1..C_K\}$ predstavlja klasno obeležje instance l_i. Neka je takođe dat dovoljno velik skup neanotiranih primera $U = \{u_i\}$, gde u_i predstavlja instancu skupa U. Neka su instance skupova L, U i T opisane istim skupom obeležja X, pri čemu ne postoji ili nije poznat prirodan način podele skupa X na dva zasebna skupa obeležja (pogleda), zbog čega direktna aplikacija ko-trening algoritma nije moguća. Međutim, primenom jednog od modela predstavljenih u ovoj disertaciji, *Algoritam Statistike Slučajnih Podela* ili *Integracija Višestrukih Ko-treninranih Klasifikatora* moguće je iskoristiti prednosti ko-trening algoritma za obučavanje kvalitetnog modela na osnovu anotiranih i neanotiranih primera i u slučaju kada podela obeležja skupa podataka nije definisana.

Prvi korak, zajednički za oba modela jeste kreiranje grupe nezavisnih ko-trening klasifikatora. Nakon kreiranja grupe nezavisnih ko-trenng klasifikatora primenjuje se alternativno jedan od modela *RSSalg* ili *IMCC*, koji se međusobno razlikuju po načinu kombinovanja dobijene grupe klasifikatora u cilju davanja predikcije klase za instance skupa T.

3.1 Kreiranje grupe nezavisnih ko-trening klasifikatora

Budući da je ko-trening veoma osetljiv na korišćenu podelu i ostale ulazne paramete [Nigam 2000b; Muslea 2002] možemo pretpostaviti da će primena ko-trening algoritma sa različitim ulaznim konfiguracijama na isti skup ulaznih podataka rezultovati obučavanjem ko-trening klasifikatora čije će performanse varirati i koji će praviti različite (nezavisne) greške prilikom

klasifikacije test podataka. Bazirano na ovoj pretpostavci, grupu od m različitih ko-trening klasifikatora $\{CL_i\}$, $i \in \{1,...,m\}$ generišemo kreiranjem m različitih slučajnih podela obeležja, a zatim primenjujemo ko-trening algoritam sa svakom od kreiranih podela nad istim skupom ulaznih podataka L i U. Slučajna podela obeležja se kreira tako što se svako obeležje na slučajan način dodeli jednom od pogleda, na taj način da rezultujući pogledi sadrže približno jednak broj obeležja.

3.2 Algoritam statistike slučajnih podela

Tokom obučavanja ko-treninga, unutrašnji par klasifikatora selektuje i anotira podskup instanci skupa neanotiranih instanci U. Označimo ovako uvećani obučavajući skup (formiran od inicijalno anotiranih primera skupa L i instanci skupa U anotiranih tokom ko-trening procesa) koji nastaje u toku obuke i-tog ko-trening klasifikatora korišćenjem i-te slučajne podele obeležja ($i \in \{1,...,m\}$) sa $Lres_i$.

Unutrašnji par ko-trening klasifikatora je različit za svaku korišćenu podelu (budući da je svaki obučen korišćenjem zasebnog pogleda nastalog na osnovu korišćene podele). Različiti parovi klasifikatora će, bazirano na svojoj pouzdanosti, selektovati i anotirati različit podskup instanci skupa neanotiranih primera (ovo će biti pojašnjeno kasnije u ovom odeljku na konkretnom primeru). Zbog toga će se formirani skupovi $Lres_i$, $i \in \{1,...,m\}$ međusobno razlikovati. Ista instanca može, a ne mora, biti prisutna u različitim *Lres* skupovima, a takođe se može desiti da ista instanca bude različito anotirana u različitim *Lres* skupovima.

U okviru *Algoritma statistike slučajnih podela* (Random Split statistics Algorithm, u daljem tekstu *RSSalg*) koraci za formiranje finalnog klasifikacionog modela su:

1. Integracija skupova $Lres_i$, $i \in \{1,...,m\}$ u jedinstven skup L_{int}
2. Izdvajanje podskupa najpouzdanije anotiranih instanci L_{rel} iz skupa L_{int}
3. Nadgledano obučavanje modela korišćenjem skupa L_{rel}.

1. Integracija skupova $Lres_i$

Rezultat ovog koraka je integrisani skup L_{int} koji sadrži svaku instancu e, koja se pojavljuje u najmanje jednom od skupova $Lres_i$. Za svaku ovakvu instancu se određuje:

- Broj skupova $Lres_i$ u kojima se instanca e pojavljuje, n_e:

$$n_e = |\{Lres_i \mid e \in Lres_i \, , i \in \{1..m\}\}| \tag{52}$$

- Za svaku postojeću klasu C_k, broj skupova $Lres_i$ u kojima je instanca e anotirana da pripada klasi C_k, n_{eCk}:

$$n_{eCk} = |\{Lresi \mid e \in Lresi \wedge label(e) = Ck \, , i \in \{1..m\}\}| \tag{53}$$

Svaka instanca e se dodaje u integrisani skup L_{int} i anotira njoj najčešće dodeljivanom klasom C, određenom većinskim glasanjem:

$$n_{eC} = \max(n_{eCk} \mid k \in \{1..\, m\}) \tag{54}$$

2. Izdvajanje podskupa najpouzdanije anotiranih instanci iz integrisanog skupa L_{int}

Glavni razlog nestabilnosti performansi metoda polu-nadgledanog obučavanja proističe iz veće mogućnosti dodele pogrešne anotacije neanotiranim podacima [Blum 2001; Nigam 2000a]. U slučaju *RSSalg* algoritma pogrešno anotirane instance uvode šum u obučavajući skup koji se koristi za treniranje finalnog klasifikatora, zbog čega je zarad formiranja što tačnijeg modela neophodno ukloniti što je moguće više ovakvih instanci. Iz ovog razloga, uvešćemo pojam pouzdane instance. U *RSSalg* pretpostavlja se da je instanca pouzdana (u smislu da je sa velikom verovatnoćom anotirana tačnom klasom) ako važi:

- Instanca je anotirana od strane dovoljnog broja klasifikatora CL_i;
- Većina klasifikatora CL_i koja je anotirala datu instancu se slaže u pogledu njene anotacije.

Na osnovu ove pretpostavke se za svaku instancu e koja pripada skupu L_{int} i anotirana je klasom C određuje:

- procenat pojave instance e u skupovima $Lres_i$ kao broj skupova *Lres* u kojima se instanca e pojavljuje podeljen sa m (ukupnim brojem skupova *Lres*): $e_{occ} = n_e/m$
- procenat slaganja anotacije e_{agg} tako što se broj skupova *Lres* u kojima je data instanca e anotirana klasom C podeli brojem skupova *Lres* u kojima se pojavljuje data instanca e: $e_{\text{agg}} = n_{eC}/n_e$

Neka su zadati prag pojave instance $example_{ts}$ i prag slaganja labele $label_{ts}$. Data instanca e se smatra pouzdanom ukoliko procenat pojave date instance e_{occ} prelazi zadati prag pojave instance ($e_{occ} > example_{ts}$) i procenat slaganja anotacije date instance prelazi zadati prag slaganja labele ($e_{\text{agg}} > label_{ts}$). Sve instance koje prema zadatim kriterijumima nisu pouzdane se izbacuju iz skupa L_{int}. Rezulat ovog koraka jeste skup L_{rel} koji predstavlja podskup pouzdanih instanci izdvojenih iz integrisanog skupa L_{int}.

3. Obučavanje finalnog modela

U ovom koraku se primenom odabrane tehnike nadgledanog obučavanja nad obučavajućim skupom L_{int} obučava finalni klasifikator koji predstavlja rezultat izvršavanja *RSSalg* algoritma. Na ovaj način se dobija model spreman za klasifikaciju novih (test) instanci.

RSSalg algoritam je sumarizovan u algoritmu 5.

Ulaz

- Skup kategorija $\{C_k\}$ u koje je neophodno razvrstati instance, $k \in \{1..K\}$
- Mali skup L anotiranih instanci
- Znatno veći skup U primera koji nisu anotirani
- Skup obeležja X kojima su opisani dati skupovi podataka
- Parametri ko-trening algoritma:
  - broj iteracija ko-trening algoritma k
  - veličina podskupa neanotiranih primera u'
  - za svaku od klasa $k \in \{1,...,K\}$ zadaje se parametar c_k koji definiše broj primera anotiranih klasom k koji će biti dodati u inicijalni obučavajući skup)
- Parametri RSSalg algoritma:
  - Broj slučajnih podela m.
  - Prag pojave instance E_{ts} i prag slaganja labele L_{ts}.

Treniranje modela

Inicijalizovati skup $S = \{(e, n_e, n_{eC1}, n_{eC2}, \ldots, n_{eCk})\}$ (e – instanca, n_e – broj pojave instance e u S, n_{eCk} – broj pojave instance e u skupu S pri čemu je instanca e anotirana klasom C_k)

Za svako i, i=1..m:

- Na slučajan način podeliti skup obeležja X na dva jednaka dela X_1 i X_2
- Koristeći datu podelu obeležja obučiti i-ti ko-trening klasifikator (algoritam 1). Izlaz ko-trening algoritma je $Lres_i = \{(ei, C_{ei})\}$, gde ei predstavlja instancu koja pripada skupu $Lres_i$, a C_{ei} klasu dodeljenu instanci ei od strane i-tog ko-trening klasifikatora
- Za svaki par $(ei, C_{ei}) \in Lres_i$ pronaći instancu ei u skupu S, uvećati broj pojave date instance n_{ei} za jedan i uvećati broj puta gde je instanca ei označena klasom C_{ei} (n_{eCei}) za jedan.

Inicijalizovati integrisani skup instanci i njima dodeljenih labela $L_{rel} = \{(e, C)\}$ na prazan skup. Za svaku instancu e skupa S:

- Odrediti najfrekventnije klasno obeležje C (za koje važi $n_{eC} = \max(n_{eCk} \mid k \in \{1..K\})$)
- Izračunati uniformnost dodeljenih klasnih obeležja za instancu $e_{agg} = n_{eC}/n_e$
- Izračunati procenat pojave instance $e_{occ} = n_e/n$
- Ukoliko važi $e_{agg} > L_{ts}$ i $e_{occ} > E_{ts}$, dodati (e, C) u skup L_{rel}

Koristiti skup L_{rel} za obučavanje finalnog modela M

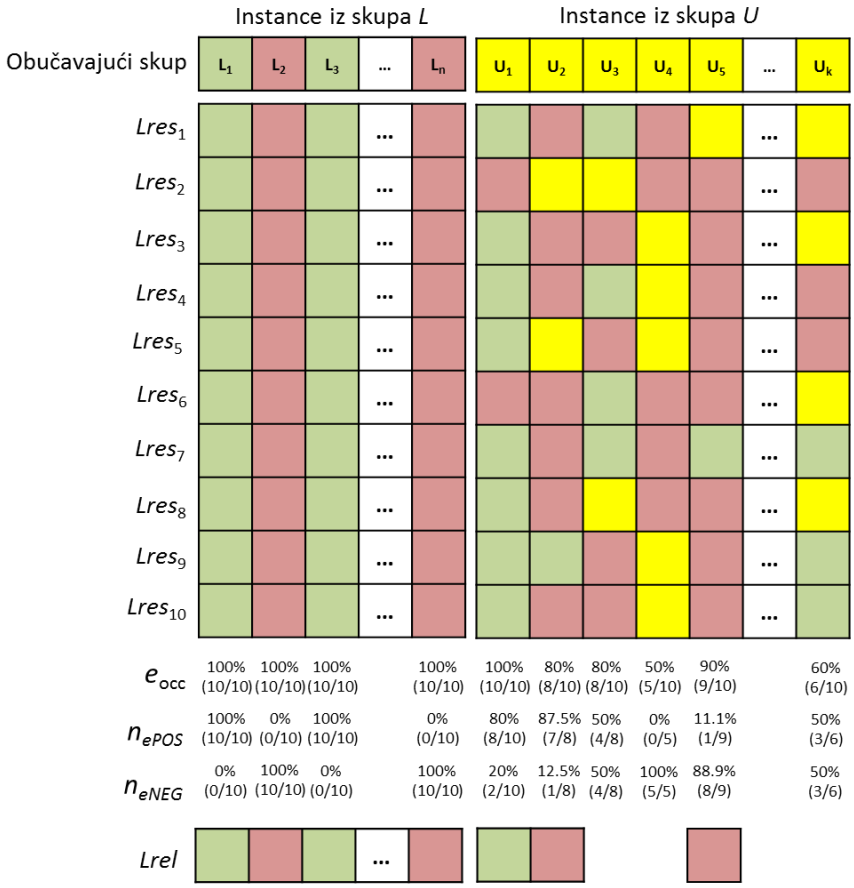
Izlaz

Model M

Algoritam 5: *RSSalg* algoritam

Slika 11 prikazuje primer formiranja obučavajućeg skupa L_{rel} u okviru *RSSalg*. U prikazanom primeru, vrednost uzeta za prag pojave instance E_{ts} je 70% (što znači da instanca može biti dodata u skup pouzdano anotiranih instanci samo ukoliko se nalazi u najmanje 70% rezultujućih ko-trening skupova $Lres_i$, $i = \{1..m\}$), a vrednost uzeta za prag slaganja labele (L_{ts}) je 80% (što znači da instanca može biti dodata u skup pouzdano anotiranih instanci samo ukoliko se

80% rezultujućih ko-trening klasifikatora koji su anotirali datu instancu slaže oko njoj dodeljene klase). Instance u ovom primeru mogu pripadati jednoj od dve klase – pozitivnoj (*POS*) i negativnoj (*NEG*). Instance koje pripadaju pozitivnoj klasi su označene zelenom bojom, instance koje pripadaju negativnoj klasi su označene crvenom bojom, dok su neanotirane instance označene žutom bojom.



Slika 11 Primer formiranja obučavajućeg skupa L_{rel}. Kolone predstavljaju instance, a redovi predstavljaju skupove podataka. Zelenom bojom prikazane su pozitivno anotirane instance, crvenom negativne, a žutom instance koje nisu anotirane.

Sve instance iz polaznog anotiranog skupa L (L_1-L_n) naći će se u skupu pouzdanih instanci L_{rel}. Ovo proizilazi iz činjenice da se u toku ko-trening procesa u polazni obučavajući skup samo dodaju novo anotirane instance, a polazne instance inicijalnog skupa se ne menjaju (ne mogu biti izuzete iz trening skupa niti im se klasno obeležje može promeniti), te će se instance skupa L naći u svim ko-trening rezultujućim skupovima $Lres_i$ (procenat pojave instance će kod ovih primera biti 100%) i, takođe, svaka ovakva instanca će imati isto klasno obeležje u svim $Lres_i$ skupovima (procenat slaganja anotacije je 100%).

Instanca U_1 je dodata u skup pouzdanih instanci L_{rel} budući da je sadržana u svim $Lres_i$ skupovima (e_{occ}=100% ≥ 70%) i da se 80% ko-trening klasifikatora koji su anotirali ovu instancu slažu u pogledu njenog klasnog obeležja (e_{agg}=*80%* ≥ 80%). Ova instanca je u skupu L_{rel} anotirana pozitivnom klasom jer ju je većina ko-trening klasifikatora anotirala pozivinom klasom. Instanca U_2 je takođe dodata u skup L_{rel} jer je sadržana u 80% skupova $Lres_i$ (≥70%) i pošto joj je procenat slaganja anotacije 87.5% (u 7 od ukupno 8 skupova u kojima je ova instanca bila anotirana, dodeljena joj je negativna klasa). Instanca U_3 je sadržana u dovoljno skupova da bi bila dodata u L_{rel} (e_{occ}=80% ≥ 70%), ali je ne smatramo pouzdanom jer ju je 50% klasifikatora anotiralo pozitivnom klasom, a 50% negativnom klasom, zbog čega ipak nije dodata u L_{rel}. Sa druge strane, 100% klasifikatora koji su anotirali instancu U_4 se slaže da ova instanca pripada negativnoj klasi, ali instancu nije anotirao dovoljan broj klasifikatora da bi se ova anotacija smatrala pouzdanom (instanca se pojavljuje u svega 5 od 10 $Lres_i$ skupova, a prag pojave instance je 70%), zbog čega ova instanca takođe nije uvršćena u skup L_{rel}. Konačno, instanca U_k nije uvršćena u skup L_{rel} jer nije anotirana u dovoljno skupova, niti se dovoljan broj klasifikatora koji ju je anotirao slaže u pogledu dodeljene anotacije.

Odabrani pragovi pojave instance i slaganja anotacije (E_{ts} i L_{ts}) u značajnoj meri utiču na performanse *RSSalg* algoritma. Odabir previsokih pragova za posledicu može imati uklanjanje informativnih, dobro anotiranih instanci koje su korisne za obučavanje finalnog modela, dok odabir previše niskih pragova može imati za posledicu dodavanje šuma u obučavajući skup koji se koristi za treniranje finalnog modela. U cilju automatske detekcije optimalnog para pragova koji maksimizuje performanse finalnog *RSSalg* modela korišćen je genetski algoritam, o čemu će biti reč u narednom odeljku.

3.2.1 Automatsko određivanje optimalnog para pragova pojave instance i slaganja anotacije

Definisanje optimalnog para pragova pojave instance i slaganja anotacije je kompleksan optimizacioni problem, zbog čega je genetski algoritam pogodan za rešavanje ovog problema.

U okviru genetskog algoritma, svaka individua sadrži dva hromozoma – jedan koji predstavlja prag pojave instance E_{ts} i jedan koji predatavlja prag slaganja anotacije L_{ts}. Oba hromozoma su predstavljena uz pomoće binarnog koda budući da je na taj način olakšana direktna primena operatora GA na vrednosti pragova.

U opštem slučaju, vrednosti pragova mogu da variraju između 0% i 100%. Konvertovanje vredosti pragova se vrši tako da se izbegne mogućnost da primenom operatora GA rešenja ispadnu iz željenog opsega. Zbog toga, ukoliko se za kodiranje vrednosti pragova koristi n bitova, broj mogućih vrednosti (2^n-1) se deli u 100 intervala kojima se dodeljuju vrednosti od 0% do 100%. Shodno

tome, za konverziju broja x koji upada u opseg (*minVal, maxVal*) u svoj n-bitni kod koristi se sledeća formula:

$$Binary[Round(\ (x - minVal)\cdot(2^n - 1)\ /\ (minVal - maxVal)\ )] \qquad (55)$$

gde *Binary* označava funkciju konverzije broja u binarni kod, a *Round* označava funkciju zaokruživanja vrednosti na najbliži ceo broj. Na primer, za kodiranje praga 55% se koristi 8-bitni kod: 10001100.

Svaka individua je predstavljena parom pragova (E_{ts}, L_{ts}). Budući da želimo da proizvedemo model najveće moguće tačnosti, logična vrednost funkcije prilagođenosti individue bi bila tačnost koju klasifikacioni model, dobijen uz pomoć datog para pragova, ima na skupu namenjenom za evaluaciju modela. Međutim, u ko-trening postavci raspolažemo samo sa malim brojem anotiranih primera, te nam nedostaju anotirane instance na kojima bi smo mogli optimizovati pragove.

Zadavanje određenog para pragova (E_{ts}, L_{ts}) rezultuje time da određene instance skupa L_{int} (koje zadovoljavaju zadate pragove) budu uvršćene u obučavajući skup finalnog modela, a da neke instance budu odbačene. Na primer, slika 11 prikazuje situaciju gde su primeri U_1, U_2 i U_5 uvršćeni u finalni obučavajući skup, dok su primeri U_3, U_4 i U_m izostavljeni iz njega. Bazirano na ideji tzv. *Out-of-bag* estimacije [Breiman 1996b], da se podaci izostavljeni iz obučavajućeg skupa prilikom primene samorazvijajućeg (*bootstrap*) metoda koriste za estimaciju modela dobijenog na osnovu instanci uvršćenih u obučavajući skup, skup za evaluaciju dobijenog modela L_{eval} predstavljaju instance skupa L_{int} koje ne prelaze zadate pragove (E_{ts}, L_{ts}), te nisu uvršćene u obučavajući skup modela L_{rel}, odnosno važi sledeći odnos:

$$L_{rel} \cup L_{eval} = L_{int}, L_{rel} \cap L_{eval} = \emptyset \qquad (56)$$

Na primer, slika 11 prikazuje situaciju gde se model treniran na obučavajućem skupu koji se sastoji od instanci $\{L_1,\ldots,L_n, U_1, U_2, U_{5,\ldots}\}$ evaluira na test skupu koji se sastoji od instanci $\{U_3, U_4,\ldots,U_m\}$.

Međutim, neki parovi pragova mogu rezultovati time da sve instance budu selektovane u L_{rel}, zbog čega skup L_{eval} može da bude prazan. Takođe, previše mali skupovi za evaluaciju (koji takođe sa velikom verovatnoćom sadrže zanačajan šum budući da su sastavljeni od najmanje pouzdanih primera) mogu rezultovati lošom procenom tačnosti formiranog modela. Zbog toga definišemo parametar test prag T_{ts} koji predstavlja minimalni broj instanci koji može sadržati evaluacioni skup individue. Ovaj broj se definiše relativno u odnosu na veličinu integrisanog skupa svih instanci L_{int}. Npr. ukoliko se test prag postavi na 20%, svaka individua koja koristi preko 80% primera u okviru L_{rel} skupa, a manje od 20% primera u okviru L_{eval} skupa se smatra za lošu jedinku čija je estimacija loša. U ovakvom slučaju, najmanje pouzdane instance se prebacuju iz obučavajućeg skupa L_{rel} u evaluacioni skup L_{eval}, tako da se u evaluacioni skupu

nađe dovoljan broj instanci (u smislu zadovoljenja praga T_{ts}). Najmanje pouzdanim instancama smatramo one koje imaju najmanji zbir procenta slaganja anotacije i procenta pojave instance.

Da bi se izračunala vrednost funkcije prilagođenosti individue, na osnovu para pragova koji predstavljaju datu individuu se skup *S* na gore opisan način deli na obučavajući skup (sastavljen od pouzdano anotiranih instanci) i test skup (sastavljen od instanci skupa *S* koje prema datim pragovima nisu pouzdane). Vrednost funkcije prilagođenosti individue se računa kao tačnost koju klasifikator obučen na datom obučavajućem skupu postiže na odgovarajućem test skupu.

Prostor pretrage je prilično velik (sve moguće kombinacije vrednosti pragova između 0 i 100) sa obzirom na potencijalno skupu evaluaciju funkcije prilagođenosti[7]. Određene delove prostora ne želimo da pretražujemo. Na primer, za binarne probleme minimalan procenat slaganja anotacije koji možemo uočiti je 50% u slučaju da polovina klasifikatora klasifikuje instancu u jednu klasu, a druga polovina u drugu klasu. Zbog toga nema smisla ispitivati kombinacije pragova koji za prag slaganja anotacije imaju vrednost manju od 50%. Možemo identifikovati minimalan prag pojave instance Ets_{min} u zabeleženom skupu *S* i minimalno slaganje anotacije Lts_{min} u zabeleženom skupu *S* i ispitivati samo kombinacije pragova koje imaju vrednosti odgovarajućih pragova veće od ovih. Sa druge strane, maksimalna vrednost praga pojave instance i praga slaganja anotacije će svakako biti 100% zbog polaznih instanci skupa *L* koji se nalaze u svim $Lres_i$ skupovima. Takođe, budući da različite kombinacije pragova mogu rezultovati selekcijom istog skupa pouzdano anotiranih instanci, u implementaciji se čuvaju izračunate vrednosti funkcije prilagođenosti za svaki uočeni skup pouzdano anotiranih instanci kako se ne bi ponovo trenirao model koji bi rezultovao istim klasifikatorom.

U fazi selekcije, vrši se selekcija individua iz generacije kojima je dozvoljeno da se razmnožavaju. Korišćena je proporcionalna (*roulette wheel*) selekcija [Back 1991]. Za rekombinaciju, u cilju stvaranja novih individual, se koristi operator uniformnog ukrštanja dve jedinke-roditelja (*bi-parental uniform crossover*) [Syswerda 1989] i operator mutacije u jednoj tački (*single-point mutation*) [Goldberg 1989]. Takođe, koristi se i elitizam, tj. očuvanje najboljih jedinki (u smislu vrednosti funkcije prilagođenosti) koje prelazi iz generacije u generaciju, kako bi se ubrzao process pretrage. U korišćenoj postavci rezervisama su dva mesta u narednoj generaciji za najbolje jednike iz prethodne generacije.

[7] Prilikom evaluacije se obučava klasifikator na potencijalno velikom broju instanci, što može da bude računarski i vremenski zahtevno.

3.3 Integracija višestrukih ko-treniranih klasifikatora

Ulaz

• Skup kategorija $\{C_k\}$ u koje je neophodno razvrstati instance, $k \in \{1..K\}$
• Mali skup L anotiranih instanci
• Znatno veći skup U primera koji nisu anotirani
• Skup neanotiranih instanci T koje je neophodno razvrstati u skup klasa $\{C_k\}$
• Skup obeležja X kojima su opisani dati skupovi podataka
• Parametri ko-trening algoritma:
  o broj iteracija ko-trening algoritma k
  o veličina podskupa neanotiranih primera u'
  o za svaku od klasa $k \in \{1,...,K\}$ broj primera c_k koji će u svakoj iteraciji biti anotirani datom klasom i dodati u inicijalni obučavajući skup)
• Parametri IMCC algoritma:
  o Broj slučajnih podela m.

Izlaz
Skup $T = \{(x_i, y_i)\}$, $i = 1..n$ gde x_i predstavlja i-tu instancu ulaznog obučavajućeg skupa D, a y_i estimaciju tačnog klasnog obeležja instance x_i.
Algoritam
Obučavanje: Za svako i, i=1..m: • Na slučajan način podeliti skup obeležja X na dva jednaka dela X_1 i X_2 • Koristeći datu podelu obeležja obučiti i-ti ko-trening klasifikator (algoritam 1). Klasifikacija novih primera: Formirati predikcioni skup podataka P koji se sastoji od instanci skupa T ($P = T$) Za svako i, i=1..m i svako $p \in P$: • Iskoristiti i-ti ko-trening klasifikator za klasifikaciju instance p. Dobijenu oznaku kategorije y_t^i dodeliti kao i-tu labelu instanci p. Rezultat prethodnog koraka je predikcioni skup $p = \{(t, y_t^1, y_t^2, \dots, y_t^m)\}$. Primeniti *GMM-MAPML* (algoritam 4) na predikcioni set P u cilju estimacije konačne kategorije y_t svake instance skupa T.

Algoritam 6: *IMCC* algoritam

Kao i u *Algoritmu Statistike Slučajnih* Podela, prvi korak u algoritmu *Integracija višestrukih ko-treniranih klasifikatora* (*Integration of Multiple Co-trained Classifiers, IMCC*) jeste kreiranje grupe nezavisnih ko-trening klasifikatora (odeljak 3.1). Svaki od m klasifikatora iz grupe klasifikatora daje svoju predikciju labele za svaku instancu skupa T. Na ovaj način se formira predikcioni skup $P = \{(t, y_t^1, \dots, y_t^m)\}$, gde t predstavlja instance skupa T, a $y_t^i \in \{C_1..C_K\}$ predstavlja klasno obeležje dodeljeno instanci t od strane i-tog ko-trening klasifikatora. Pretpostavka da su ko-trening klasifikatori nezavisni u smislu grešaka koje prave prilikom klasifikacije novih instanci nam omogućava

da ih tretiramo kao grupu nezavisnih anotatora i da stoga možemo primeniti *GMM-MAPML* algoritam (odeljak 1.4.3) za estimaciju pravog klasnog obeležja svake instance skupa *T* na osnovu višestrukih anotacija nepoznatog kvaliteta.

Sumarizacija *GMM-MAPML* algoritma je prikazana u algoritmu 6.

3.4 Motivacija za korišćenje predloženih modela

Motivacija koja stoji iza predloženih modela jeste da neanotirani podaci mogu biti od velike koristi za tehnike učenja sa grupom hipoteza [Zhou 2009].

Neophodni i dovoljni uslovi da grupa klasifikatora ima veće performanse od bilo kojih od svojih individualnih klasifikatora jesu:

- klasifikatori koji pripadaju datoj grupi moraju biti budu raznovrsni (*diverse*);
- svaki klasifikator zasebno mora imati dobre performanse.

Prvi uslov je ispunjen ukoliko se za obučavanje klasifikatora koristi algoritam čije su performanse veoma osetljive na ulazne postavke [Dietterich 2000]. Ko-trening algoritam je veoma osetljiv na korišćenu podelu obeležja [Nigam 2000b; Muslea 2002]. Zbog toga, primenom ko-treninga na isti skup podataka, ali koristeći različite slučajne podele obeležja, možemo očekivati da ćemo dobiti grupu raznovrsnih klasifikatora.

Eksperimentalne evaluacije [Nigam 2000b; Chan 2004; Koprinska 2007] su pokazale da ko-trening primenjen sa slučajnom podelom obeležja ima dobre performanse ukoliko se skup podataka odlikuje velikom redudantnošću obeležja, pa možemo da očekujemo da će na ovakvim skupovima podataka biti ispunjen i drugi uslov neophodan da bi grupa formiranih klasifikatora imala visoke performanse.

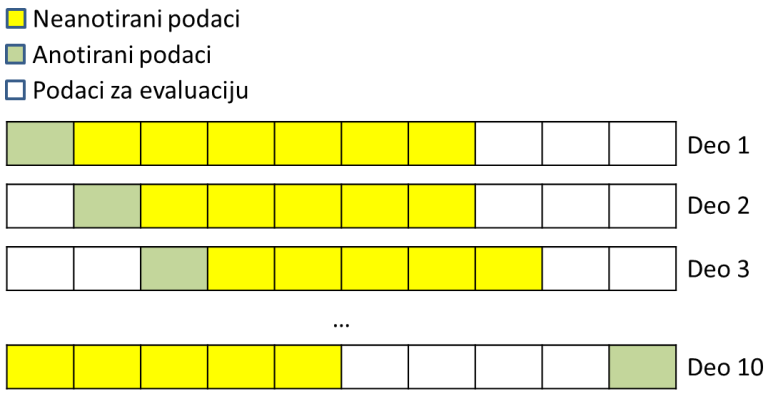
Kao i u *bagging* tehnici, *RSSalg* i *IMCC* metode integrišu grupu klasifikatora i eksploatišu razlike u lokalno različitim modelima sa ciljem podizanja tačnosti. Prva faza i kod *bagging* tehnike i kod *RSSalg* i *IMCC* metoda je kreiranje tzv. samorazvijajuće (bootstrap) replike originalnog skupa podataka. Skup replika se u *bagging* tehnici kreira slučajnim izvlačenjem instanci iz skupa podataka, nakon čega se na replikama primenjuju obučavajući algoritmi radi formiranja klasifikatora. Ukoliko raspolažemo sa veoma malim brojem anotiranih primera, na ovaj način nije moguće napraviti značajno različite replike originalnog skupa koje istovremeno sadrže dovoljno podataka na kojima bi smo mogli obučiti kvalitetne klasifikatore. Međutim, u slučaju da imamo pristup dovoljno velikoj količini neanotiranih podataka, moguće je primeniti *RSSalg* ili *IMCC* metodologiju koje replike originalnog trening skupa sastavljaju koristeći i anotirane i neanotirane podatke, primenom ko-trening algoritma. Na ovaj način (ukoliko je primena ko-treninga uspešna) dobijamo grupu raznolikih klasifikatora još boljeg kvaliteta u odnosu na klasifikator treniran nadgledanim obučavanjem na originalnom skupu podataka.

Dobijeni klasifikatori se u *bagging* tehnici kombinuju primenom većinskog glasanja – instanci za koju je neophodno utvrditi klasnu pripadnost se dodeljuje klasa oko koje se slaže najveći broj formirane grupe klasifikatora. U *RSSalg* modelu primenom glasanja kombinuju predikcije pojedinačnih klasifikatora za primere koji potiču iz neanotiranog skupa podataka, a anotirani su tokom ko-trening procesa. Ovako anotirani primeri neizostavno sadrže i šum klasifikacije. Međutim, budući da se one anotiraju tehnikom sličnom *bagging* tehnici za koju je pokazano da je stabilna i otporna na šum klasifikacije [Dietterich 2000], rezultat bi trebao da bude skup značajno tačnije anotiranih instanci u odnosu na anotacije dodeljene datim instancama u pojedinačnim skupovima-replikama. Sa druge strane, u *IMCC* algoritmu se, kao i u *bagging* tehnici, klasa nove instance određuje tako što svaki klasifikator glasa kojoj klasi data instanca pripada. Za razliku od *bagging* metode, gde se vrši prosto većinsko glasanje, u *IMCC* modelu se za svaki od klasifikatora određuje njegova senzitivnost i specifičnost za različite grupe podataka. Senzitivnost i specifičnost oslikavaju koliko je dati klasifikator kvalitetan na kojoj grupi podataka. Prilikom određivanja klase neanotirane instance uzima se u obzir kojoj grupi podataka ona pripada, nakon čega se vrši težinsko glasanje (težina klasifikatora zavisi od kvaliteta klasifikatora za datu grupu podataka) grupe klasifikatora.

RSSalg i *IMCC* imaju sličnosti i sa *RSM* metodom. Naime, u *RSM* metodi se na slučajan način biraju podskupovi obeležja koji se koriste za konstrukciju pojedinačnih klasifikatora grupe. Slično, *RSSalg* i *IMCC* kreiraju raznolikost u grupi klasifikatora manipulacijom skupa obeležja. Najveći nedostatak *RSM* metode je što ne postoji garancija da odabrani podskup obeležja sadrži dovoljno diskriminativnih informacija, što rezultuje lošim klasifikatorom koji može da ošteti tačnost formirane grupe klasifikatora [Garcia-Pedrajas 2008]. U *RSSalg* i *IMCC* ovaj nedostatak je donekle ublažen time što se kroz ko-trening proces koriste informacije celog skupa obeležja, iako se takođe ne može garantovati da će pojedinačni ko-trening klasifikator sa slučajnom podelom obeležja imati dobre performanse. U *IMCC* algoritmu i ovaj nedostatak je donekle prevaziđen estimacijom senzitivnosti i specifičnosti pojedinačnog klasifikatora koje se koriste prilikom formiranja finalne predikcije.

4 Evaluacija modela

U cilju evaluacije razvijenih modela, usvojena je metodologija evaluacije korišćena u [Feger 2006]. Predložena metodologija evaluacije predstavlja modifikovanu desetostruku unakrsnu validaciju (*10-fold-cross validation*). U standardnoj proceduri desetostruke unakrsne validacije se obučavajući skup deli na 10 delova približno jednake veličine dobijenih stratifikovanim (*stratified*) uzorkovanjem. Stratifikovano uzorkovanje predstavlja uzorkovanje prilikom koga se vodi računa da u dobijenom uzorku postoji ista distribucija klasa koja postoji u originalnom skupu podataka (iz koga se vrši uzorkovanje). U svakoj od iteraciji desetostruke unakrsne validacije se model obučava na 90% podataka, a testira na preostalih 10% podataka. U ko-trening algoritmu se koristi samo mala količina označenih i neoznačenih primera, tako da bi korišćenje standardne desetostruke unakrsne validacije rezultovalo time da su mnogi primeri isključeni i iz obučavajućeg skupa i iz skupa za evaluaciju. Radi boljeg iskorišćenja podataka kojima raspolažemo, bolje je uvećati veličinu skupa test podataka, čime se unapređuje evaluacija klasifikatora, dok se kvalitet datog klasifikatora ne smanjuje u značajnoj meri. Kako bi se ovo postiglo, skup podataka je podeljen na 10 stratifikovanih delova. U svakoj iteraciji desetostruke unakrsne validacije se koristi drugi deo (10% podataka) u okviru koga se vrši slučajna selekcija neophodne količine anotiranih podataka. Ostatak primera iz tog dela, kao i podaci iz 5 njemu susednih delova se tretiraju kao neanotirani primeri. Ostatak podataka (iz preostala 4 dela) se koristi kao skup test podataka. Ovim pristupom se u svakoj iteraciji 60% podataka koristi za treniranje, a preostalih 40% podataka za testiranje klasifikatora. Svaki od delova skupa podataka se koristi tačno jednom za izbor anotiranih podataka, 5 puta biva uvršćen u neoznačene podatke, i 4 puta se koristi kao deo skupa test podataka. Opisani proces prikazan je na slici 12.



Slika 12 Desetostruka unakrsna validacija korišćena za testiranje ko-trening algoritma.

Kao mera performanse treniranih klasifikatora korišćena je tačnost (*accuracy*). Ova mera je široko korišćena kao mera performanse za ko-trening algoritam i računa se po formuli:

$$accuracy = \frac{tp + tn}{tp + fp + tn + fn} \tag{57}$$

gde *tp* i *tn* predstavljaju broj primera ispravno klasifikovanih kao pozitivnih, odnosno negativnih, respektivno, a *fp* i *fn* predstavljaju broj primera pogrešno klasikikovanih kao pozitivnih, odnosno negativnih, respektivno. Drugim rečima, tačnost se računa kao odnos broja tačno klasifikovanih primera i ukupnog broja primera.

4.1 Skupovi podataka za evaluaciju

Cilj ove teze jeste razvoj automatskog sistema za klasifikaciju koji će biti robustan i primenljiv na širokom spektru domena. Kako bi se proverila robustnost sistema, za evaluaciju je odabrano 17 skupova podataka različitih po dimezionalnosti, broju instanci i redudantnosti obeležja u koje je neophodno razvrstati zapise.

Odabrana su 3 skupa podataka prirodnog jezika (*natural language datasets*) koja su i ranije korišćena u svrhu evaluacije ko-trening algoritma: WebKB-Course [Blum 1998; Nigam 2000a], News2x2 [Feger 2006; Nigam 2000b] i Ling Spam korpus [Feger 2006, Chan 2004]. Značajna karakteristika skupova podataka prirodnog jezika jeste visok nivo redudantnosti obeležja[8] [Joachims 2001]. Zbog toga, najbolje performanse *RSSalg* i *IMCC* metodologija očekujemo upravo na ovim skupovima podataka.

Sem njih, sa UCI repozitorijuma za mašinsko učenje[9] je odabrano je 14 skupova podataka nad kojima je potrebno izvršiti binarnu klasifikaciju. Ovi skupovi podataka su takođe korišćeni od strane prethodnih autora za evaluaciju ko-treninga [Du 2010; Zhou 2005a; Huang 2010; Hady 2008a; Li 2007]. UCI skupovi podataka su obično kreirani pažljivim manuelnim odabirom obeležja i zbog toga na ovim skupovima podataka ne očekujemo visok nivo redudantnosti.

Korišćeni skupovi podataka i njihove najvažnije osobine su predstavljeni u tabeli 1. Prva 4 skupa podataka (WebKB, LingSpam, News2x2 i Spambase sa UCI repozitorijuma) su skupovi prirodnog jezika, a sledećih 13 skupova

[8] Prirodan jezik je sam po sebi redudantan. U njemu se često javljaju dvosmislenosti u cilju čijeg razrešenja se često uvodi redudantnost – značenje reči je često zavisno od konteksta. Takođe, često se dešava da individualne reči gledano zasebno nemaju poseban značaj, ali ga dobijaju kada su posmatrane kao fraza (npr. „biti ili ne biti“). Zbog postojanja korelacije među rečima (koje se često koriste kao obeležja skupova podataka prirodnog jezika) rezultujući skupovi podataka se često odlikuju visokim nivoom redudantnosti.

[9] UCI repozitorijum za mašinsko učenje se nalazi na adresi http://archive.ics.uci.edu/ml/datasets.html

podataka su UCI skupovi podataka koje je neophodno razvrstati u jednu od dve kategorije.

Skup podataka	Dim	|L|	L_{acc}	|All|	All_{acc}	Optimlan dobitak
WebKB	400	5/5	78.6	138/492	96.4	17.8
Spambase	57	2/1	67.7	1672/1087	79.6	11.9
LingSpam	400	5/5	80.1	288/1447	88.9	8.8
News2x2	400	5/5	81.1	600/600	89.6	8.5
Hepatitis	19	1/1	61.7	19/73	84.8	23.1
Kr-vs-kp	36	6/5	65.6	1001/916	87.2	21.6
Credit-g	20	1/1	53.6	420/180	74.1	20.5
Heart-statlog	13	3/2	65.7	90/72	80.5	14.8
Cylinder-bands	39	2/3	58.4	136/187	72.9	14.5
Sonar	60	1/1	55.5	66/58	68.8	13.3
Ionosphere	34	5/3	70.1	135/75	83.1	13.0
Breast-cancer	9	1/1	59.0	51/120	71.7	12.7
Credit-a	15	4/5	69.3	184/229	81.5	12.2
Tic-tac-toe	9	3/6	58.8	199/375	70.7	11.9
Breast-w	9	1/2	86.3	144/274	97.4	11.1
Mushroom	22	3/3	84.7	2524/2349	95.3	10.6
Diabetes	8	2/1	64.8	300/160	75.0	10.2

Tabela 1 Sumarizacija najvažnijih osobina skupova podataka korišćenih za evaluaciju. Notacija: **Dim** – dimenzionalnost, odnosno, broj instanci skupa podataka; **|L|** - veličina malog anotiranog skupa podataka u formatu broj instanci koje pripadaju pozitivnoj klasi/broj instanci koje pripadaju negativnoj klasi; $\mathbf{L_{acc}}$ – tačnost koju postiže NB klasifikator obučen na malom skupu anotiranih podataka *L*; **|All|** - veličina celog obučavajućeg skupa (t.j. zbir brojeva instanci malog anotiranog skupa *L* i neanotiranog skupa *U*), takođe u formatu broj instanci koje pripadaju pozitivnoj klasi/broj instanci koje pripadaju negativnoj klasi; $\mathbf{All_{acc}}$ –tačnost koju postiže NB klasifikator obučen na celom obučavajućem skupu *All* (t.j. na obučavajućem skupu koji se sastoji od anotiranih instanci skupa *L* i instanci neanotiranog skupa *U* kojima je dodeljena tačna anotacija); **Optimalan dobitak** – procena mogućeg pobojšanja tačnosti u odnosu na polaznu tačnost L_{acc}. Računa se po formuli $All_{acc} – L_{acc}$.

4.1.1 Pretprocesiranje tekstualnih skupova podataka

Radi konstrukcije modela za klasifikaciju teksta neophodno je nestruktuirani tekst prevesti u strukturiran oblik. U ovoj tezi korišćena je najrasprostranjenija reprezentacija tekstualnih dokumenata – model "vreće reči" (*bag-of-words model*). U ovoj reprezentaciji se svaki dokument predstavlja kao vektor težina termina (reči ili fraza) koji se javljaju u njemu, pri čemu se ne uzima u obzir pozicija reči u rečenici, niti mogući međusobni odnosi mežu rečima. Najrasprostranjenije korišćena mera koja se koristi za izračunavanje težine (značajnosti) reči jeste takozvana *tf-idf* metoda (*term frequency-inverse document frequency*) [Salton 1988]. Vrednost *tf-idf* mere izračunava se na sledeći način. Neka je $tf_{i,j}$ frekvencija termina *i* u dokumentu *j* (*term frequency*) koja se definiše na sledeći način:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{58}$$

gde je $n_{i,j}$ broj pojavljivanja termina t_i u dokumentu d_j, a $\sum_k n_{k,j}$ predstavlja sumu broja pojavljivanja svih termova u dokumentu d_j. Neka je idf_i inverzna frekvencija termina t_i u kolekciji dokumenata (*inverse document frequency*). Ova mera predstavlja meru specifičnosti termina za dokument i računa se kao

kao logaritam podele broja svih dokumenata sa brojem dokumenata koje sadrže term:

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|}. \tag{59}$$

Tada se *tf-idf* mera računa na sledeći način:

$$(tf - idf)_{i,j} = tf_{i,j} \times idf_i. \tag{60}$$

Visoka vrednost *tf-idf* mere se postiže visokom frekvencijom termina (u datom dokumentu) i niskom frekvencijom dokumenata koje sadrže termin u celoj kolekciji dokumenata (kako bi se isključili uobičajeni termini). *Tf-idf* mera je takođe nezavisna od klasnog obeležja primera, i kao takva ne narušava postavku ko-treninga.

Ovaj način reprezentacije tekstualnih dokumenata okarakterisan je veoma velikom dimenzionalnošću skupa podataka jer se svaki termin u korpusu tretira kao obeležje. Visoka dimenzionalnost rezultuje retkim (*sparse*) podacima što može da degradira performanse i rezultuje dugotrajnim vremenom izvršavanja klasifikacionih algoritama. Zbog toga neophodno izvršiti selekciju obeležja, odnosno minimizaciju broja obeležja uz zadržavanje što je moguće više informacija. U ovom cilju korišćene su sledeće tehnike pretprocesiranja prilikom određivanja termina korišćenih u modelu[10]: konverzija teksta u mala slova, tokenizator teksta (*string tokenization*), uklanjanje 319[11] čestih reči engleskog jezika primenom tehnike uklanjanja stop reči i korenovanje reči[12], izvedeno pomoću Porterovog algoritma [Porter 1980].

Nakon primene opisanih tehnika pretprocesiranja teksta, u cilju dalje redukcije dimenzionalnosti, primenjen je i postupak pretprocesiranja baziran na frekvenciji pojavljivanja termina (*term*) u dokumentu (*document frequency*). Frekvencija pojave termina u dokumentu se računa kao broj dokumenata u kojima se dati termin pojavljuje i pretpostavlja da su frekventniji termini značajniji od manje frekventnih termina. Ova mera ne zahteva poznavanje klasnog obeležja, što je prednost u ko-trening postavci gde nam je poznato klasno obeležje samo malog broj primera, a takođe ima linearnu kompleksnost.

[10] Pored ovde navedenih postupaka, u slučaju WebKB skupa podataka, iz teksta su takođe uklonjeni telefonski brojevi, sekvence cifara, datumi i karakteri koji nisu alfa-numerički, budući da je utvrđeno da ovi karakteri nemaju značajan uticaj na predikciju klase dokumenta.

[11] Lista od 319 reči engledskog jezika je definisana od strane univerziteta u Glazgovu http://ir.dcs.gla.ac.uk/resources.html

[12] Korenovanje (*stemming*) je proces formiranja morfoloških korena reči na osnovu reči iz teksta. Cilj jeste da se grupa različitih istog korena reprezentuje svojim korenom, čime se postiže efektivno smanjenje dimenzionalnosti prostora termina.

Upotrebom ove mere selektovano je 200 najznačajnih obeležja za svaki od pogleda.

Opisani postupak reprezentacije tekstualnih dokumenata je preuzet iz [Feger 2006] radi što preciznijeg poređenja predloženih rešenja sa *maxInd* metodologijom.

4.2 Vrednosti parametara korišćene u evaluaciji

Inicijalni mali skup anotiranih primera *L* odabran je tako da njegova distribucija klasa odgovara stvarnoj distribuciji klasa u celom skupu podataka (osim u slučaju *LingSpam* skupa podata, gde se radi lakšeg poređenja koristi isti broj instanci kao u [Feger 2006]). Ovo uobičajen način odabira skupa *L* u srodnim radovima gde se testiraju performanse ko-treninga [Blum 1998; Feger 2008; Du 2010]. Takođe, prilikom odabira veličine skupa anotiranih primera *L*, vođeno je računa o tome da ko-trening ima prostora da unapredi performanse, slično kao u [Du 2010], t.j. odabran je mali inicijalni obučavajući skup tako da je optimalan dobitak (definisan u tabeli 1) dovoljno velik (10-20%). Broj anotiranih primera za svaki skup podataka je izlistan u tabeli 1.

Kao u i [Feger 2006], veličina podskupa neanotiranih primera *u'* (*unlabeled pool*, odeljak 1.3.4) je 50, a broj iteracija ko-trening algoritma *k* je 20. Broj slučajnih podela osobina *m* korišćenih u okviru *RSSalg* i *IMCC* algoitama je 50 za tekstualne skupove podataka, a 30 za *UCI* skupove podataka. Za svaki skup podataka korišćen za evaluaciju, brojevi instanci *p* i *n* anotiranih pozitivnom i negativnom klasom, respektivno, koji se u svakoj iteraciji dodaju obučavajućem skupu *L* su odabrani tako da se očuvava stvarna distribucija klasa skupa podataka, kao što je preporučeno u [Blum 1998].

Bazični klasifikator koji se koristi u okviru ko-trening algoritma je Naivni Bajesov klasifikator (NB). Izbor ovog klasifikatora zasnovan je na visokoj tačnosti koje postiže na skupovima podataka korišćenim za evaluaciju, kao i na njegovoj brzini (što je važan faktor zbog velike kompleksnosti *RSSalg* i *IMCC* algoritama). Takođe, za naivni Bajesov klasifikator je pokazano da postiže dobre performanse na problemima tekstualne kategorizacije [Lewis 1993][Ting 2011], na kojima očekujemo najviše performanse *RSSalg* i *IMCC* algoritama zbog velike redundancije tekstualnih skupova podataka. Konačno, ovaj klasifikator je najčešće korišćen za evaluaciju ko-trening algoritma [Feger 2006].

Za process optimizacije pragova pojave instance slaganja anotacije je korišćena sledeća postavka GA: svaka generacija se sastojala od 50 jedinki, a ukupan broj generacija (broj iteracija GA) je bio 10. Verovatnoća zamene bitova u uniformmom ukrštanju je bila 0.3, verovatnoća mutacije je bila 0.002 i korišćen je elitizam. Vrednost uzeta za test prag je T_{ts} 20%. Svi navedeni parametri su empirijski odabrani.

4.3 Poređeni algoritmi

U procesu evaluacije su poređeni sledeće ko-trening postavke:

1. Ko-trening sa priprodnom podelom obeležja (za skupove podataka WebKB, News2x2 i LingSpam gde je ova podela poznata), u daljem tekstu ***Natural***.
2. Ko-trening sa slučajnom podelom obeležja, u daljem tekstu ***Random***. U cilju realnijeg poređenja, ko-trening je primenjen na *m* različitih slučajnih podela (istih podela korišćenih u okviru *RSSalg* i *IMCC* algoritama), a rezultati prikazani za *Random* predstavljaju prosek dobijenih performansi.
3. Postavka većinskog glasanja (*Majority Vote*, u daljem tekstu ***MV***) u kojoj se klasa test instance određuje većinskim glasanjem *m* nezavisnih ko-trening klasifikatora nastalih korišćenjem *m* različitih slučajnih podela (istih podela korišćenih u okviru *RSSalg* i *IMCC* algoritama). Ovaj način kombinovanja klasifikatora je sličan *bagging* tehnici, te služi kao osnovni model (*baseline*) sa kojima poredimo kombinovanja dobijenih klasifikatora koje su korišćene u *RSSalg* i *IMCC* algoritmu.
4. Ko-trening sa podelom obeležja dobijenom uz pomoć *maxInd* metodologije [Feger 2006]. Dosadašnji rezultati su pokazali da ova postavka ne daje uvek najbolje rezultate ukoliko se u okviru ko-treninga koristi NB klasifikator, te da može imati značajno veće performanse ukoliko se kombinuje sa drugim klasifikatorima kao što su radijalne bazne funkcije (*Radial Basis Function, RBF*) mreže i mašine potpornih vektora (*Support Vector Machine, SVM*) [Feger 2006]. Zbog toga je ova postavka na svakom skupu podataka primenjena koristeći svaki od klasifikatora NB, SVM i RBF u okviru ko-treninga, a prikazane su najbolje postignute performanse. U daljem tekstu će ova postavka biti označena sa $\boldsymbol{MaxInd_{best}}$.
5. Algoritam statistike slučajnih podela (*Random Split Statistics algorithm, RSSalg***)** koji je jedan od modela predstavljenih u ovoj tezi (odeljak 3.2). Algoritam statistike slučajnih podela čiji su pragovi L_{ts} i E_{ts} optimizovani uz pomoć GA opisanog u odeljku 3.2.1 će u daljem tekstu biti označen kao ***RSSalg***.
6. Algoritam statistike slučajnih podela čiji su pragovi L_{ts} i E_{ts} optimizovani uz pomoć GA koji kao funkciju prilagođenosti koristi tačnost modela evaluiranog na ručno anotiranom test skupu (koji se koristi za evaluaciju konačnog modela), u daljem tekstu $\boldsymbol{RSSalg_{best}}$. Ovu postavku možemo posmatrati kao gornju granicu performansi koje bi *RSSalg* mogao dostići na test podacima ukoliko bi smo odabrali idealne pragove L_{ts} i E_{ts}.
7. Integracija višestrukih ko-trening klasifikatora (*Integration of Multiple Co-trained Classifiers, IMCC*) koji je jedan od modela predstavljenih u ovoj tezi (odeljak 3.3). U daljem tekstu će ova postavka biti označena kao ***IMCC***.

4.4 Rezultati i diskusija

U tabeli 2 prikazana je tačnost koju postižu testirane ko-trening postavke na svim skupovima podataka. Na najviše skupova podataka (13 od ukupno 17) IMCC postavka se pokazala kao najuspešnija, a sledeća najbolja postavka je $RSSalg_{best}$.

Datasets	Natural	Random	MV	$MaxInd_{best}$	RSSalg	RSSalg best	IMCC
WebKB	87.2±6.7	84.2±7.6	87.7±3.5	78.3±9.1	87.3±5.1	**90.7±3.3**	88.6±1.0
LingSpam	70.3±13.3	76.6±8.5	81.1±7.4	83.9±1.1	88.5±7.0	91.1±5.9	**95.3±0.5**
News2x2	82.9±4.6	80.0±7.0	86.3±2.4	76.2±12.8	89.1±3.1	**90.6±1.8**	85.7±0.7
Spambase		67.8±15.8	77.4±4.7	68.9±8.2	78.2±7.7	81.5±4.1	**81.5±0.5**
Hepatitis		80.3±8.0	83.3±4.3	80.8±7.9	82.6±3.5	**86.5±3.4**	85.8±5.1
Kr-vs-kp		54.4±5.0	55.3±4.5	60.1±6.2	58.3±5.7	67.1±4.2	**79.1±1.0**
Credit-g		62.0±5.5	64.4±5.5	68.1±1.8	62.7±6.8	70.2±0.7	**70.7±1.8**
Heart-statlog		79.4±8.2	81.8±2.0	80.8±4.5	81.1±3.2	83.3±2.2	**85.2±2.6**
Cylinder-bands		52.5±5.2	52.9±6.5	56.3±5.7	54.3±4.8	61.6±2.5	**65.9±1.9**
Sonar		54.9±6.0	56.5±5.5	56.7±9.1	56.7±8.0	61.2±5.8	**62.4±3.8**
Ionosphere		69.4±12.6	73.1±4.9	78.3±7.7	74.6±7.3	**79.6±5.8**	75.2±3.0
Breast-cancer		66.7±6.1	68.2±4.5	67.5±5.4	67.0±6.8	70.4±5.3	**73.9±2.2**
Credit-a		69.2±15.0	73.4±11.0	76.1±2.6	72.0±8.4	77.6±4.8	**79.6±1.2**
Tic-tac-toe		61.5±3.2	63.2±2.5	62.0±1.7	61.6±3.0	64.1±2.9	**70.5±1.5**
Breast-w		96.8±0.8	96.9±0.7	96.7±0.7	96.5±1.0	97.5±0.4	**97.6±0.4**
Mushroom		88.2±3.2	89.1±1.0	88.4±1.3	88.6±1.4	89.2±0.9	**89.9±0.7**
Diabetes		61.4±7.3	64.1±3.3	65.3±1.1	63.9±3.7	67.7±1.8	**71.7±2.1**

Tabela 2 Poređenje alternativnih ko-trening postavki. U tabeli je prikazana postignuta tačnost i standardna devijacija (u procentima) dobijena u proceduri stratifikovane 10-struke unakrsne validacije na svakom od skupova podataka. Postavka *Natural* je primenljiva samo na prva tri skupa podataka (WebKB, LingSpam i News2x2) za koje je poznata prirodna podela obeležja. Najveća postignuta tačnost za svaki od skupova podataka je označena masnim slovima (bold).

Radi dalje karakterizacije rezultata, takođe je primenjen *Friedman*-ov statistički test praćen *Bergman-Hommel*-ovim post hoc testovima, opisanim u odeljku 4.4.1. Ovo je standardna procedura za poređenje više različitih klasifikatora na više različitih skupova podataka preporučena u [Demšar 2006] i [Garcia 2008].

4.4.1 Primenjeni statistički testovi

Statistička evaluacija eksperimentalnih rezultata se smatra esencijalnim delom validacije novih metoda mašinskog učenja. Tipično, performanse razvijenog modela se porede sa performansama alternativnih modela na različitim skupovima podataka. Najčešće korišćena mera performanse je tačnost (*accuracy*). Cilj je da se statistički verifikuje da li razvijena metoda zaista pokazuje veće performanse od testiranih alternativa [Demšar 2006]. Na primer,

dobijene varijacije tačnosti mogu poticati iz slučajne selekcije test podataka koji se koriste za evaluaciju algoritama. Na određenom test skupu, uzorkovanom na slučajan način, jedan klasifikator može da pokazuje bolje performanse od drugog klasifikatora iako ovi klasifikatori na celokupnoj populaciji imaju identične performanse. Ovo je naročito izraženo u slučaju malih test skupova. Drugi izvor varijacije može biti i selekcija obučavajućeg skupa, naročito kod nestabilnih algoritama gde male promene u obučavajućem skupu[13] uzrokuju velike promene u obučenom klasifikatoru. Treći izvor varijacije može proisticati iz unutrašnje slučajnosti koja postoji u obučavajućem algoritmu. Na primer, kod ko-treninga se na slučajanim izvlačenjem instanci iz skupa neanotiranih podataka kreira manji podskup (*pool*) *u'* na kome se primenjuju unutrašnji klasifikatori. Konačno, varijabilnost može da proističe i iz slučajnog šuma – ukoliko je fiksiranoj frakciji od η test instanci dodeljena pogrešna labela, ni jedan obučavajući algoritam ne može da rezultuje greškom klasifikacije manjom od η [Dietterich 1998].

Za statističku analizu dobijenih rezultata korišćen je *Friedman*-ov test [Friedman 1937; Friedman 1940], praćen odgovarajućim post hoc testovima, preporučen za poređenje jednog ili više klasifikatora primenjenih na više skupova podataka [Demšar 2006][Garcia 2008]. *Friedman*-ov test predstavlja neparametrizovani ekvivalent analizi varijanse *ANOVA* [Fisher 1959].

Prvi korak u ovom postupku je da se primeni *Friedman*-ov test radi eventualnog odbacivanja nulte hipoteze da svi metodi imaju jednake performanse. Ukoliko se ova nulta hipoteza odbaci, dalje se nastavlja sa post hoc testovima. Kao post hoc test ovde je korišćen *Bergman-Hommel*-ov test[14]. Za sve testove korišćen je nivo značajnosti od $\alpha = 0.05$, što je standardna vrednost praga koja se koristi za određivanje značajnosti hipoteze [Shazmeen 2013].

Statističke procedure primenjene su za poređenje modela čije su performanse izlistane u tabeli 2. U ovoj tabeli su poređeni modeli koji su navedeni u odeljku 4.3, a čije su performanse izlistane u tabeli 2. Sa ovim modelima su u tabeli 2 poređeni i NB modeli obučeni nad malim inicijalnim anotiranim skupom *L* i velikim anotiranim skupom *All*. Performanse ovih modela su prikazane u tabeli 1 u kolonama L_{acc} i All_{acc}, respektivno. Jedini model koji je izostavljen iz ove analize je *Natural*, budući da je bilo suviše malo skupova podataka na kojima je bilo moguće izmeriti njegove performanse.

Prvi korak u *Friedman*-ovom testu jeste rangiranje algoritama na svakom od skupova podataka odvojeno, pri čemu algoritam najboljih performansi dobija rang 1, sledeći najbolji rang 2, itd. U slučaju da dva

[13] U smislu odabira instanci obučavajućeg skupa.

[14] Softver korišćen za *Friedman*-ov i *Bergman-Hommelov* test predstavljen je u radu [Garcia 2008] i dostupan je na adresi http://sci2s.ugr.es/keel/multipleTest.zip.

algoritma imaju jednake performanse, dodeljuju im se prosečni rangovi. Proces dodele rangova je prikazan u tabeli 3. Iz tabele 3 vidimo da je redosled algoritama (od najboljeg ka najgorem): All_{acc}, *IMCC*, $RSSalg_{best}$, *MV*, *RSSalg* i $MaxInd_{best}$ (koji su izjednačeni po rangu), L_{acc} i, na kraju, *Random*.

Skup podataka	**Random**	**MV**	**MaxInd best**	**RSSalg**	**RSSalg best**	**IMCC**	**L_{acc}**	**All_{acc}**
WebKB	84.2 (6)	87.7 (4)	78.3 (7)	87.3(5)	90.7 (2)	88.6 (3)	78.6 (8)	96.4 (1)
LingSpam	76.6 (8)	81.1 (6)	83.9 (5)	88.5(4)	91.1 (2)	95.3 (1)	80.1 (7)	88.9 (3)
News2x2	80.0 (7)	86.3 (4)	76.2 (8)	89.1(3)	90.6 (1)	85.7 (5)	81.1 (6)	89.6 (2)
Spambase	67.8 (7)	77.4 (5)	68.9 (6)	78.2(4)	81.5 (1.5)	81.5 (1.5)	67.7 (8)	79.6 (3)
Hepatitis	80.3 (7)	83.3 (4)	80.8 (6)	82.6 (5)	86.5 (1)	85.8 (2)	61.7 (8)	84.8 (3)
Kr-vs-kp	54.4 (8)	55.3 (7)	60.1 (5)	58.3(6)	67.1 (3)	79.1 (2)	65.6 (4)	87.2 (1)
Credit-g	62.0 (7)	64.4 (5)	68.1 (4)	62.7(6)	70.2 (3)	70.7 (2)	53.6 (8)	74.1 (1)
Heart-statlog	79.4 (7)	81.8 (3)	80.8 (5)	81.1(4)	83.3 (2)	85.2 (1)	65.7 (8)	80.5 (6)
Cylinder-bands	52.5 (8)	52.9 (7)	56.3 (5)	54.3(6)	61.6 (3)	65.9 (2)	58.4 (4)	72.9 (1)
Sonar	54.9 (8)	56.5 (6)	56.7 (4.5)	56.7(4.5)	61.2 (3)	62.4 (2)	55.5 (7)	68.8 (1)
Ionosphere	69.4 (8)	73.1 (6)	78.3 (3)	74.6(5)	79.6 (2)	75.2 (4)	70.1 (7)	83.1 (1)
Breast-cancer	66.7 (7)	68.2 (4)	67.5 (5)	67.0(6)	70.4 (3)	73.9 (1)	59.0 (8)	71.7 (2)
Credit-a	69.2 (8)	73.4 (5)	76.1 (4)	72.0(6)	77.6 (3)	79.6 (2)	69.3 (7)	81.5 (1)
Tic-tac-toe	61.5 (7)	63.2 (4)	62.0 (5)	61.6(6)	64.1 (3)	70.5 (2)	58.8 (8)	70.7 (1)
Breast-w	96.8 (5)	96.9 (4)	96.7 (6)	96.5(7)	97.5 (2)	97.6 (1)	86.3 (8)	97.4 (3)
Mushroom	88.2 (7)	89.1 (4)	88.4 (6)	88.6(5)	89.2 (3)	89.9 (2)	84.7 (8)	95.3 (1)
Diabetes	61.4 (8)	64.1 (6)	65.3 (4)	63.9(7)	67.7 (3)	71.7 (2)	64.8 (5)	75.0 (1)
Prosečan rang	7.24	4.94	5.26	5.26	2.38	2.09	6.94	1.88

Tabela 3 Rangovi dodeljeni pojedinačnim algoritmima. Kolone u tabeli odgovaraju primenjenim postavkama, a redovi odgovaraju skupovima podataka na kojima su date postavke primenjene. Vrednosti u ćelijama tabele predstavljaju tačnost odgovarajuće postavke na odgovarajućem skupu podataka i rang koji data postavka ima na datom skupu podataka u formatu „tačnost (rang)“. Na primer, ako se posmatra prvi red i prva kolona tabele, u datoj ćeliji je prikazano da *Random* postavka na WebKB skupu posataka postiže tačnost od 84.2% i da je na WebKB skupu podataka ova postavka šesta po rangu.

Višestruko poređenje ko-trening metoda na svim skupovima podataka korišćenjem *Friedman*-ovog testa je odbacilo nultu hipotezu da u proseku svi algoritmi imaju jednake performanse (hi-kvadrat distribucija sa 7 stepeni slobode χ^2_F = 129.52 sa p-vredošću od 1.09E-10). Budući da je nulta hipoteza odbačena, izvršen je *Begrman-Hommel*-ov test za detekciju važnih razlika u parovima klasifikatora.

U ovom postupku moramo izračunati i sortirati odgovarajuću statistiku i *p*-vrednosti. Kao što je objašnjeno u [Demšar 2006], statistika za poređenje *i*-tog i *j*-tog klasifikatora se računa prema sledećoj formuli:

$$z = \frac{(R_i - R_j)}{\sqrt{\frac{k(k+1)}{6N}}}, \tag{61}$$

gde R_i i R_j predstavljaju prosečan rang klasifikatora *i*, odnosno klasifikatora *j*, respektivno, izračunat u *Friedman*-ovom testu, *k* predstavlja broj poređenih klasifikatora, a *N* predstavlja broj skupova podataka korišćenih u poređenju. Za korišćenih 17 skupova podataka, standardna greška u poređenju parova klasifikatora je $SE = \sqrt{k(k+1)/6N} = \sqrt{8 \cdot 9/6 \cdot 17} = 0.840$. U tabeli 4 su prikazani rezultati post hoc poređenja urađenog prema *Bergman-Hommelovoj* proceduri sa nivoom značajnosti od $\alpha = 0.05$ i prilagođenim *p*-vrednostima [Wright 1992].

i	hipoteza	$z=(R_0-R_i)/SE$	p
1	Random vs. All_{acc}	6.371274382	1.87E-10
2	Random vs. IMCC	6.126225368	9.00E-10
3	L_{acc} vs. All_{acc}	6.021204361	1.73E-09
4	Random vs. $RSSalg_{best}$	5.776155347	7.64E-09
5	IMCC vs. L_{acc}	5.776155347	7.64E-09
6	$RSSalg_{best}$ vs. L_{acc}	5.426085326	5.76E-08
7	RSSalg vs. All_{acc}	4.025805242	5.68E-05
8	$MaxInd_{best}$ vs. All_{acc}	4.025805242	5.68E-05
9	RSS_{alg} vs. IMCC	3.780756227	1.56E-04
10	$MaxInd_{best}$ vs. IMCC	3.780756227	1.56E-04
11	MV vs. All_{acc}	3.640728218	2.72E-04
12	RSSalg vs. $RSSalg_{best}$	3.430686206	6.02E-04
13	$MaxInd_{best}$ vs. $RSSalg_{best}$	3.430686206	6.02E-04
14	MV vs. IMCC	3.395679204	6.85E-04
15	MV vs. $RSSalg_{best}$	3.045609183	0.0023221
16	Random vs. MV	2.730546164	0.0063229
17	MV vs. L_{acc}	2.380476143	0.0172903
18	Random vs. $MaxInd_{best}$	2.345469141	0.0190032
19	Random vs. $RSSalg_{best}$	2.345469141	0.0190032
20	$MaxInd_{best}$ vs. L_{acc}	1.99539912	0.0459994
21	RSSalg vs. L_{acc}	1.99539912	0.0459994
22	$RSSalg_{best}$ vs. All_{acc}	0.595119036	0.551764
23	MV vs. RSSalg	0.385077023	0.70018
24	MV vs.$MaxInd_{best}$	0.385077023	0.70018
25	Random vs. L_{acc}	0.350070021	0.726286
26	$RSSalg_{best}$ vs. IMCC	0.350070021	0.726286
27	IMCC vs. All_{acc}	0.245049015	0.806418
28	$MaxInd_{best}$ vs. RSSalg	3.17E-15	1

Tabela 4 Familija hipoteza sortirana po *p*-vrednosti prema *Bergman-Hommel*-ovoj proceduri za nivo značajnosti α=0.05.

U tabeli 5 sumirano je poređenje više klasifikatora na više skupova podataka.

Algorithm	Random	MV	$MaxInd_{best}$	RSSalg	$RSSalg_{best}$	IMCC	L_{acc}	All_{acc}
Random		0	0	0	1	1	0	1
MV	0		0	0	1	1	0	1
$MaxInd_{best}$	0	0		0	1	1	0	1
RSSalg	0	0	0		1	1	0	1
$RSSalg_{best}$	1	1	1	1		0	1	0
IMCC	1	1	1	1	0		1	0
L_{acc}	0	0	0	0	1	1		1
All_{acc}	1	1	1	1	0	0	1	

Tabela 5 Značajne razlike u parovima klasifikatora dobijene primenom *Bergman-Hommel*-ove procedure kao post hoc procedure. Značajna razlika među klasifikatorima je obeležena sa 1, dok 0 označava da među klasifikatorima ne postoji značajna razlika.

Iz tabele 5 možemo zaključiti sledeće:

- Od poređenih ko-trening postavki najbolje performanse su pokazale postavke $RSSalg_{best}$ i *IMCC*, koje su se takođe statistički izjednačile sa postavkom All_{acc}
- Ostale ko-trening postavke *Random*, $MaxInd_{best}$ i *RSSalg* su se međusobno statistički izjednačile, a takođe su izjednačene sa postavkom L_{acc}.

Na osnovu ovih rezultata vidimo da su prema izvršenim statističkim testovima, posmatrano na svim testiranim skupovima podataka, predloženi algoritmi $RSSalg_{best}$ i *IMCC* dostigli željenu tačnost: njihove performanse su jednake performansama klasifikatora treniranim nadgledanim obučavanjem na značajno većem skupu anotiranih podataka. Međutim, postavke *Random*, $MaxInd_{best}$ i *RSSalg*, posmatrano na svim skupovima podataka, nisu uspele da unaprede performanse polaznog klasifikatora.

$RSSalg_{best}$ postavka, koja predstavlja gornju granicu performansi koje želimo da postignemo *RSSalg* postavkom je dala obećavajući rezultat – ukoliko bismo odabrali odgovarajuće vrednosti pragova, *RSSalg* postavka bi pokazala zavidne performanse. Međutim, budući da je $RSSalg_{best}$ postavka pobelila *RSSalg* postavku, kao i da *RSSalg* postavka nije unapredila polazni klasifikator, možemo zaključiti da predloženi način optimizacije pragova nije efektivan na svim skupovima podataka.

Friedman-ov statistički test nam je pomogao da uporedimo više klasifikatora nad više skupova podataka. Međutim, od značaja bi bilo i razumeti koliko je koji algoritam efektivan nad pojedinim grupama skupova podataka (kao što su npr. redudantni skupovi prirodnog jezika ili manje redudantni UCI skupovi) koji imaju međusobno slične karakteristike. Na primer, jedan od rezultata *Friedman*-ovog testa jeste da *Random* postavka nije uspela da unapredi performanse polaznog klasifikatora posmatrano ukupno na svim testiranim

skupovima podataka. Budući da se većina skupova podataka na kojima su rešenja testirana ne odlikuje velikom redudantnošću (neophodnom da bi *Random* postavka bila uspešna) nemoguće je na ovaj način proceniti da li ipak postoji podskup domena na kome bi ova postavka bila uspešna. Očekujemo da će *Random* postavka imati značajno bolje performanse posmatrano na grupi skupova podataka prirodnog jezika koji bi trebali da se odlikuju velikom redudantnošću. Zbog ovoga bi bilo korisno identifikovati pojedinačne skupove podataka na kojima je data postavka uspešna u cilju određivanja generalnih svojstava podataka na kojima je moguće uspešno primeniti datu postavku.

Za poređenje parova klasifikatora nad istim skupom podataka korišćen je *ANOVA* (*analysis of variance*) *F*-test za ponovljena merenja [Fisher 1959]. Ovo je standardna statistička metoda za testiranje razlika na dva ili više srodnih uzoraka. Kao srodni uzorci ovde se koriste performanse klasifikatora izmerene više puta na istom skupu podataka, korišćenjem više slučajnih podela datog skupa na obučavajući i test skup[15]. Nulta hipoteza koja se testira je „svi klasifikatori imaju iste performanse i da su opservirane razlike u performansama slučajnost". Agregirani rezultati *F*-testa dobijeni poređenjem performansi svih parova u smislu pobeda/izjednačenje/gubitak su razmatrani zasebno na skupovima podataka prirodnog jezika i na manje redundantnim *UCI* skupovima podataka. Ovi rezultati su prikazani u tabelama 6 i 7, respektivno.

	Natural	**Random**	**MV**	**MaxInd**	**RSSalg**	**RSSalg best**	**IMCC**	**L_{acc}**	**All_{acc}**
Natural		0/2/1	0/2/1	1/1/1	0/1/2	0/1/2	0/2/1	1/2/0	0/0/3
Random	1/2/0		0/3/1	1/2/1	0/1/3	0/0/4	0/1/3	1/3/0	0/0/4
MV	1/2/0	1/3/0		3/1/0	0/2/2	0/2/2	0/2/2	3/1/0	0/1/3
MaxInd	1/1/1	1/2/1	0/1/3		0/1/3	0/0/4	0/0/4	0/4/0	0/0/4
RSSalg	2/1/0	3/1/0	2/2/0	3/1/0		0/4/0	1/2/1	4/0/0	0/3/1
RSSalg best	2/1/0	4/0/0	2/2/0	4/0/0	0/4/0		1/2/1	4/0/0	0/3/1
IMCC	1/2/0	3/1/0	2/2/0	4/0/0	1/2/1	1/2/1		4/0/0	2/0/2
L_{acc}	0/2/1	0/3/1	0/1/3	0/4/0	0/0/4	0/0/4	0/0/4		0/0/4
All_{acc}	3/0/0	4/0/0	3/1/0	4/0/0	1/3/0	1/3/0	2/0/2	4/0/0	

Tabela 6 Agregiran broj **pobeda/izjednačenja/gubitaka** svakog para klasifikatora na **pojedinačnim skupovima podataka prirodnog jezika**. Na primer, 1/2/0 u redu *MV* i koloni *Natural* označava da je *MV* klasifikator imao značajno bolje performanse od *Natural* klasifikatora na 1 skupu podataka prirodnog jezika, da su se performanse *MV* i *Natural* klasifikatora statistički izjednačile na 2 skupa podataka prirodnog jezika, i da je *MV* pokazao statistički gore performanse od *Natural* na 0 skupova podataka prirodnog jezika.

[15] Za svaki skup podataka napravljeno je više podela na obučavajući/test skup, nakon čega su sve poređene postavke primenjene na dobijene podele (kako bi u jednom poređenju dve postavke imale isti obučavajući/test skup).

	Random	MV	MaxInd	RSSalg	RSSalg best	IMCC	L_{acc}	All_{acc}
Random		0/13/0	0/9/4	0/12/1	0/4/9	0/2/11	7/4/2	0/2/11
MV	0/13/0		0/13/0	0/12/0	0/7/6	0/4/9	7/4/2	0/2/11
MaxInd	4/9/0	0/13/0		1/11/0	0/8/5	0/3/10	6/6/1	0/2/8
RSSalg	1/12/0	0/13/0	0/12/1		0/7/6	0/3/10	4/7/1	0/3/10
RSSalg best	9/4/0	6/7/0	5/8/0	6/7/0		0/9/4	12/1/0	0/5/8
IMCC	11/2/0	9/4/0	10/3/0	10/3/0	4/9/0		13/0/0	2/3/8
L_{acc}	2/4/7	2/4/7	1/6/6	1/7/4	0/1/12	0/0/13		0/0/13
All_{acc}	11/2/0	11/2/0	10/3/0	10/3/0	8/5/0	8/3/2	13/0/0	

Tabela 7 Agregiran broj **pobeda/izjednačenja/gubitaka** svakog para klasifikatora na **pojedinačnim UCI skupovima podataka**. Na primer, 1/12/0 u redu *RSSalg* i koloni *Random* označava da je *RSSalg* klasifikator imao značajno bolje performanse od *Random* klasifikatora na 1 UCI skupu podataka, da su se performanse *RSSalg* i *Random* klasifikatora statistički izjednačile na 12 UCI skupova podataka, i da je *RSSalg* pokazao statistički lošije performanse od *Random* na 0 UCI skupova podataka.

Na osnovu tabela 2 i 6 možemo izvesti sledeće zaključke koji se odnose na skupove prirodnog jezika, izdvojenih u posebnu grupu za analizu zbog uvećane redudancije obeležja koja pogoduje *Random*, *RSSalg* i *IMCC* postavkama:

1. *RSSalg* i $RSSalg_{best}$ postavke su pobedile *Natural* postavku na dva od ukupno tri skupa podataka na kojima je bilo moguće testirati *Natural* postavku, a na preostalom skupu podataka (WebKB) ove dve postavke su se izjednačile. Ovo nam daje indikaciju da je *RSSalg* postavka bolja od *Natural* za skupove podataka sa većom redudancijom obeležja. *IMCC* postavka je pobedila *Natural* postavku na jednom skupu podataka (LingSpam), dok su se na dva preostala skupa podataka ove postavke izjednačile. Dakle, i *IMCC* postavka postiže bolje ili iste rezultate od *Natural* postavke na skupovima podataka prirodnog jezika.
2. Na tri od ukupno četiri testirana skupa podataka, *RSSalg* i *IMCC* postavke su pobedile *Random* postavku, a na preostalom skupu podataka (WebKB) su se sa njom izjednačile. Ovo nam daje indikaciju da su *RSSalg* i *IMCC* postavke bolje od *Random* postavke za ovakve skupove podataka.
3. Na dva od četiri skupa podataka (LingSpam i News2x2) *RSSalg* postavka je pobedila *MV* postavku, dok su se na preostala dva skupa podataka ove dve postavke izjednačile. Slično je i sa *IMCC* postavkom koja je pobedila *Random* postavku na LingSpam i Spambase skupovima podataka, dok se na ostalim skupovima podataka sa njom izjednačila. Dakle, načini kombinovanja predikcija primenjeni u *RSSalg* i *IMCC* postavci su uspešni na ovakvim skupovima podataka jer daju iste ili bolje rezultate od većinskog glasanja.
4. Na ovim skupovima podataka *RSSalg* postavka je postigla bolje performanse od $MaxInd_{best}$ postavke – pobedila ju je na tri skupa podataka, a na četvrtom

(LingSpam) ove postavke su se statistički izjednačile. *IMCC* i $RSSalg_{best}$ postavke su na sva četiri skupa pokazala bolje performanse od $MaxInd_{best}$ postavke.

5. Na sva četiri skupa podataka postavke *RSSalg* i $RSSalg_{best}$ su se statistički izjednačile. Kako $RSSalg_{best}$ predstavlja gornju granicu performansi *RSSalg* postavke, možemo zaključiti da je primenjena metoda automatske detekcije pragova uspešna na skupovima prirodnog jezika. *RSSalg* i $RSSalg_{best}$ se na skupovima prirodnog jezika odlikuju sličnim performansama kao *IMCC* postavka.
6. Na svim skupovima podataka *RSSalg*, $RSSalg_{best}$ i *IMCC* postavka su pobedile L_{acc} postavku.
7. Nijedna od ostalih alternativa nije uspela da pobojša polazni klasifikator na svim testiranim skupovima prirodnog jezika:
   a. U izvedenim eksperimentima, *Natural* postavka je unapredila polazni klasifikator na samo jednom skupu podataka (WebKB). Potrebno je napomenuti da je ova postavka uspešno primenjena i na News2x2 [Nigam 2000b; Feger 2008] i na *LingSpam* skupove podataka [Feger 2008]. U [Feger 2008] su korišćeni drugačiji unutrašnji klasifikatori ko-treninga (RBF) i izvršeno je poređenje sa drugačijim polaznim klasifikatorom (RBF), dok su za NB klasifikator performanse slične ovde predstavljenima [Feger 2008]. Razlike u odnosu na rezultate prikazane u [Nigam 2000b] potiču od razlika u eksperimentalnoj postavci (u smislu veličine skupa *L* i ko-tening parametara).
   b. $MaxInd_{best}$ postavka se na svim skupovima podataka statistički izjednačila sa L_{acc}. Potrebno je napomenuti da autori u [Feger 2008] za *MaxInd* postavku prijavljuju bolji porast u performansama u odnosu na polazni klasifikator. Međutim, u radu je porast performansi je meren u odnosu na polazni *RBF* klasifikator, koji prikazuje slabije performanse od *NB* klasifikatora obučenog na istom inicijalnom skupu *L*, koji je korišćen za poređenje u ovde prikazanim eksperimentima.
   c. *Random* postavka je pokazala nešto bolje performanse od L_{acc}, ali i od *Natural* postavke. Ovo se slaže sa prethodnim rezultatima da *Random* postavka može biti uspešna ukoliko postoji dovoljno redudancije u podacima [Nigam 2000b].
8. *RSSalg*, $RSSalg_{best}$ i *IMCC* postavke imaju slične performanse kao All_{acc} postavka, odnosno postižu performanse klasifikatora treniranog nadglednim obučavanjem na značajno većem anotiranom skupu podataka. Nijedna alternativna metoda nije uspela da dostigne ove performanse (*MV* postavka se sa Acc_{all} postavkom izjednačila samo na jednom skupu podataka - Spambase). Drugim rečima, korišćenjem *RSSalg* ili *IMCC* postavke je sa svega nekoliko anotiranih primera postignuta tačnost koju bi klasifikator imao ukoliko je obučen na značajno većem obučavajućem skupu. Na primer, na *LingSpam* skupu podataka se polazeći od svega 10 anotiranih instanci dostigla tačnost klasifikatora obučenog na 1735 anotiranih instanci.

9. *MV* postavka je pokazala bolje performanse od $MaxInd_{best}$ postavke (pobedila ju je na tri skupa podataka, a izjednačila se sa njom na preostalom skupu podataka, LingSpam). Takođe, *MV* postavka se izjednačila sa *Natural* i *Random* postavkama na većini skupova podataka, ali je, za razliku od ovih postavki, pokazala bolje performanse u odnosu na L_{acc} postavku (nije uspela da pobojša polazni klasifikator jedino na LingSpam skupu podataka).

Na osnovu tabela 2 i 7 možemo izvesti sledeće zaključke o manje redudantnim UCI skupovima podataka:

1. *RSSalg* postavka ima veoma slične performanse kao *Random* postavka. Pobedila ju je na samo jednom skupu podataka (Kr-vs-kp), dok su na ostalim skupovima podatka izjednačene.
2. Na svih 13 skupova podataka *RSSalg* i *MV* imaju slične performanse, što znači da na UCI skupovima podataka način odabira pragova nije dao zadovoljavajuće rezultate.
3. *RSSalg* i $MaxInd_{best}$ postavka imaju veoma slične performanse.
4. *RSSalg* postavka se izjednačila sa $RSSalg_{best}$ postavkom na sedam skupova podataka, a izgubila od nje na šest skupova podataka. Ovaj rezultat takođe ukazuje da način odabira pragova korišćen u *RSSalg* postavci nije dovoljno uspešan na manje redudantnim skupovima podataka.
5. *RSSalg* postavka je uspela da unapredi polazni klasifikator L_{acc} na svega četiri skupa podataka, a degradirala je njegove performanse na jednom skupu. Dakle, u slučaju skupova podataka koji se odlikuju nižom redudancijom obeležja, *RSSalg* postavka nije pouzdana.
6. *RSSalg* postavka se izjednačila sa All_{acc} postavkom na tri testirana skupa podataka.
7. *RSSalg* postavka se pokazala dosta lošija od *IMCC* postavke.
8. Generalno, na ovoj grupi skupova podataka, *Random*, $MaxInd_{best}$, *MV* i *RSSalg* postavka pokazuju slične performanse i ne uspevaju da pobojšaju polazni klasifikator (L_{acc}) na većini UCI skupova.
9. Sa druge strane, gornja granica performansi *RSSalg* – $RSSalg_{best}$ postavka pokazuje prilično dobre performanse i na UCI skupovima podataka: značajno je bolja od *Random* postavke i prikazuje nešto bolje rezultate od $MaxInd_{best}$ i *MV* postavke. Ova postavka je takođe uspela da unapredi polazni klasifikator L_{acc} na svim skupovima podataka sem jednog i uspela da dostigne performanse All_{acc} klasifikatora na pet skupova podataka. Dakle, *RSSalg* metodologija ima potencijala, ali je neophodna bolja tehnika automatske selekcije pragova za skupove podataka manje redudantnosti.
10. Na UCI skupovima podataka se najbolje pokazala *IMCC* postavka. Ova postavka je pokazala bolje performanse od svih ostalih testiranih ko-trening postavki i unapredila performanse polaznog klasifikatora na svim testiranim skupovima podataka. Takođe, pobedila je All_{acc} postavku na dva testirana skupa podataka i izjednačila se sa njom na tri skupa podataka.

Generalno, u ovde izvedenim eksperimentima, na skupovima podataka prirodnog jezika najbolje performanse su pokazala predložena rešenja *RSSalg* i *IMCC*, koja su uspela da pobojšaju polazni klasifikator i čak dostignu performanse klasifikatora obučenog na mnogo većoj količini podataka. Sledeća najbolja postavka je *MV*, nakon čega slede *Random* i *Natural* postavke. $MaxInd_{best}$ postavka se pokazala neuspešnom na skupovima podataka prirodnog jezika, budući da ni na jednom testiranom skupu nije pobojšala performanse polaznog klasifikatora.

Na UCI skupovima podataka najbolje performanse je pokazala *IMCC* postavka, a druga najbolja postavka je bila $RSSalg_{best}$. Ostale postavke su se pokazale kao neefektivne na ovakvim skupovima podataka, budući da nisu uspele da unaprede performanse u odnosu na polazni klasifikator.

Dakle, generalno *IMCC* postavka se pokazala kao najpouzdanija. Za *RSSalg* postavku se pokazalo da je potrebno unaprediti proceduru za automatsku optimizaciju pragova (konkretno, potrebno je unaprediti način evaluacije funkcije prilagođenosti) jer se ova metoda pokazala neefektivna na manje redudantnim skupovima podataka. Pokazano je da i na ovoj grupi skupova podataka *RSSalg* metoda ima veliki potencijal jer uz pravilan izbor pragova postiže prilično veliku tačnost.

4.5 Analiza uticaja vrednosti parametara na performanse rešenja

U idealnom slučaju, robustan algoritam bi trebao imati što je moguće manje parametara za optimizaciju i pokazivao bi dobre performanse na širokom skupu domena. Mnogi algoritmi polu-nadgledanog obučavanja, ukjučujući i ko-trening, su veoma osetljivi na korišćene vrednosti parametara [Goldberg 2009], zbog čega je njihova primena u praksi ograničena.

Parametar koji najviše utiče na performanse ko-treninga jeste korišćena podela obeležja [Nigam 2000b][Muslea 2002]. Predložena rešenja, *IMCC* i *RSSalg*, eliminišu potrebu za definisanjem podele obeležja. Međutim, ova rešenja uvode sopstvene parametre – broj korišćenih slučajnih podela obeležja *m*, a, u slučaju *RSSalg* algoritma, veliki uticaj na performanse imaju i odabrani pragovi za eliminaciju nepouzdano anotiranih primera. U ovom odeljku je u odeljku 4.5.1 testiran uticaj broja korišćenih slučajnih podela *m*, a u odeljku 4.5.2 je analiziran uticaj odabranih pragova za eliminaciju nepouzdano anotiranih rešenja na performanse *RSSalg* algoritma.

U trenutku izvođenja ovih eksperimenata implementacija *GMM-MAPML* algoritma, nepohodnog za testiranje *IMCC* postavke više nije bila javno dostupna, a na osnovu specifikacije date u radu [Zhang 2011] nije je bilo moguće reimplementirati. Zbog toga nije analiziran uticaj različitih parametara na performanse *IMCC* postavke. U dosadašnjim rezultatima je pokazano da se

dodavanjem više nezavisih anotatora performanse *GMM-MAPML* algoritma pobojšavaju [Zhang 2011], zbog čega bismo očekivali i porast performansi *IMCC* algoritma sa porastom broja slučajnih podela. Pretpostavka koju uvodi *GMM-MAPML* algoritam jeste da su anotatori nezavisni. Nezavisnost anotatora (klasifikatora) se u *IMCC* postavci postiže korišćenjem različitih slučajnih podela obeležja. Ukoliko bi ova pretpostavka bila narušena, odnosno, ukoliko bi više ko-treninga rezultovalo sličnim klasifikatorima, moguće je da bi došlo do degradacije u performansi *IMCC* algoritma. U ovom slučaju, očekivali bismo i degradaciju u performansi *MV* postavke bazirane na većinskom glasanju, budući da je za uspeh grupe klasifikatora potrebno da klasifikatori grupe budu međusobno raznovrsni [Dietterich 2000]. Pokazano je i da *GMM-MAPML* postiže performanse iste ili bolje od većinskog glasanja [Zhang 2011]. Zbog ovoga, očekujemo da ne dolazi do degradacije u performansama *IMCC* postavke ukoliko ne uočimo degradaciju u performansama *MV* postavke.

U odeljku 4.5.3 analiziran je je uticaj postojanja redudantnosti u skupu podataka na performanse modela u cilju određivanja grupe skupova podataka na koje je moguće pouzdano primeniti predstavljene modele.

Ranije studije su pokazale da su performanse ko-treninga osetljive na korišćene vrednosti parametara [Nigam 2000b; Pierce 2001; Ng 2003]. Nedostatak principijelnog načina za odabir vrednosti parametara je istaknut kao veliki nedostatak ko-treninga [Ng 2003]. U [Mihalcea 2004] je empirijski pokazano da postoji veliki raskorak u performansama ko-treninga primenjenog sa optimalnim vrednostima parametara (optimizovanim kroz merenja na test skupu) i empirijski odabranim parametrima. Međutim, ova studija ne otkriva nikakve jasne veze među parametrima koji dovode do maksimizacije performansi – u nekim slučajevima klasifikatorima odgovara „agresivnije“ povećanje skupa anotiranih podataka novim primerima, dok drugima više odgovaraju „mali koraci“.

U ovom odeljku biće analizirana i robustnost *RSSalg* postavke u odnosu na vrednosti ko-trening parametara koje veoma utiču na performanse samog ko-treninga: broj iteracija ko-treninga (odeljak 4.5.4) i veličina rasta anotiranog skupa u svakoj iteraciji ko-treninga (odeljak 4.5.5). Ispitivanje uticaja kombinacija svih ovih parametara je veoma iscrpan zadatak, naročito ako se uzme u obzir vremenska kompleksnost *RSSalg* algoritma. Zbog toga će ovde će biti analizirani uticaji pojedinačnih parametara pri čemu će vrednosti ostalih parametara biti fiksirane. Ovo će nam ipak dati uvid u robustnost ili osetljivost *RSSalg* postavke na vrednosti korišćenih parametara.

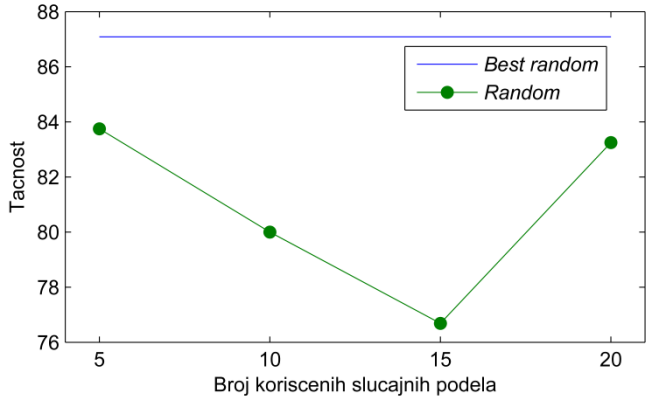
Za testiranje je odabrano nekoliko od skupova podataka opisanih u odeljku 4.1. Iz grupe skupova prirodnog jezika odabrani su News2x2 i LingSpam skupovi podataka. Iz grupe UCI skupova podataka odabrani su sledeći skupovi podataka: Breast-cancer, Diabetes, Hepatitis i Cylinder-bands. Skupovi podataka Breast-cancer i Diabetes se odlikuju izuzetno malim brojem

obeležja, tako da je na njima moguće isprobati sve moguće podele skupa obeležja na dva podskupa približno iste veličine. Skupovi podataka Hepatitis i Diabetes imaju veći broj mogućih podela. Na skupovima podatka podataka Hepatitis i Breast-cancer *RSSalg* je pokazao relativno dobre performanse, dok je na Diabetes i Cylinder-bands skupovima podatka *RSSalg* pokazao izjednačenenje u performansama sa polaznim klasifikatorom i degradaciju performansi u odnosu na polazni klasifikator, respektivno.

4.5.1 Uticaj broja korišćenih slučajnih podela

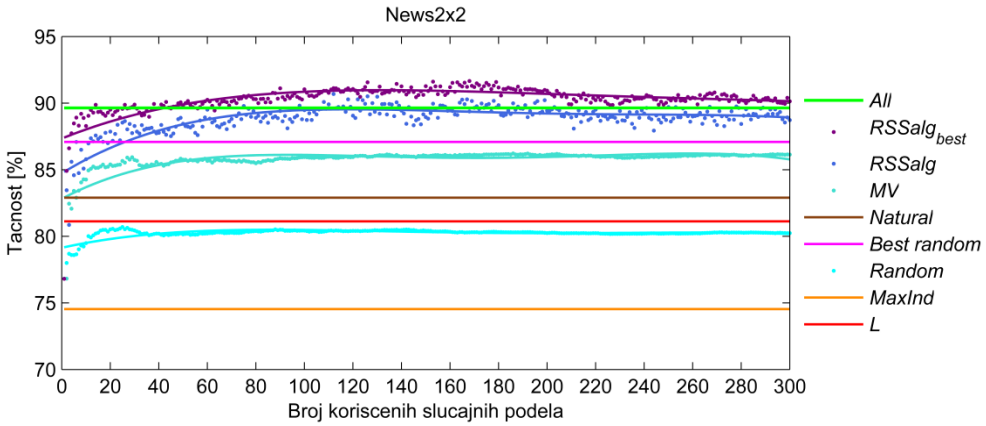
U ovom odeljku analiziran je uticaj broja korišćenih slučajnih podela (odnosno broja ko-trening klasifikatora) m na performanse *RSSalg* algoritma. Eksperiment je izvršen tako što je za svaku novu podelu novi klasifikator dodavan u grupu već postojećih ko-trening klasifikatora. Odnosno, grupa klasifikatora dobijena sa vrednošću parametra $m = 22$ se od grupe klasifikatora dobijene sa vrednošću paramerta $m = 21$ razlikuje u samo jednom klasifikatoru. Ovo nam omogućava da donekle isključimo efekat slučajnosti koji bi mogao uslediti iz toga što smo za dva uzastopna merenja koristili klasifikatore značajno različitog kvaliteta. Maksimalna testirana vrednost parametra m je bila 300, osim u slučaju Breast-cancer i Diabetes skupova podatka, gde je usled malog broja obeležja bilo moguće testirati sve moguće slučajne podele na poglede približno jednakih veličina. Svi ostali parametri su u ovom eksperimentu fiksirani na vrednosti izlistane u odeljku 4.2.

Rezultati ovih eksperimenata za različite skupove podatka su grafički prikazani na grafikonima 2 – 7. Horizontalna osa na ovim graficima predstavlja broj slučajnih podela, a vertikalna osa predstavlja tačnost postignutu od strane različitih algoritama (u procentima). Analizirane su performanse ko-trening postavki izlistanih u odeljku 4.3, kao i performanse naivnog Bajesovog klasifikatora treniranog na malom inicijalnom anotiranom skupu (ova postavka je na graficima označena sa L) i performanse naivnog Bajesovog klasifikatora treniranog na značajno uvećanom anotiranom skupu sastavljenog od inicijalnog anotiranog skupa i neanotiranog skupa čijim su instancama pridružene tačne anotacije (ova postavka je na graficima označena sa *All*). Takođe, prikazane su i performanse postavke *Best random* koja predstavlja ko-trening klasifikator sa slučajnom podelom obeležja za koji je utvrđeno da ima najviše performanse od svih ko-trening klasifikatora nastalih korišćenjem slučajnih podela obeležja. Smisao zabeleženih performansi *Random* i *Best random* podele je pojašnjen na grafikonu 1.



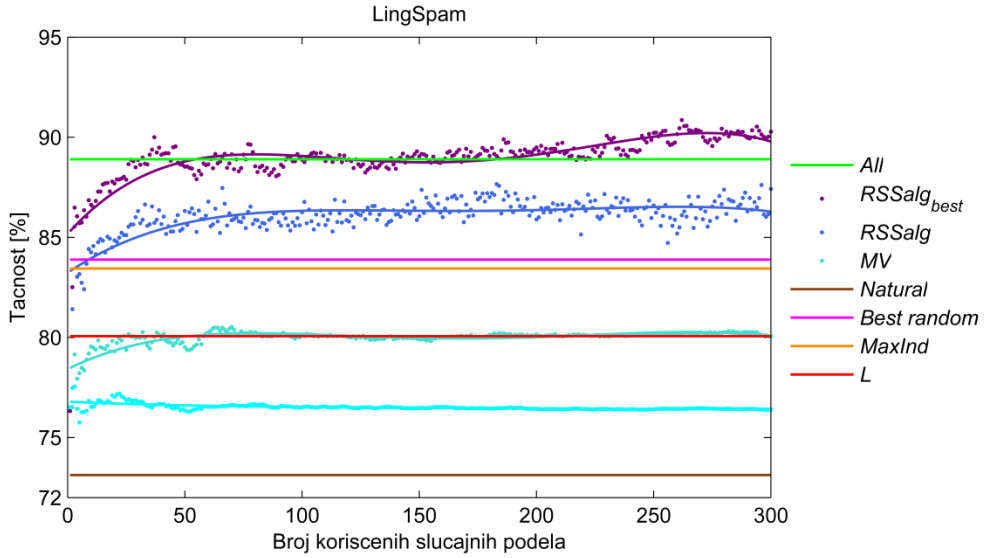
Grafikon 1 Pojašnjenje izmerenih performansi *Random* i *Best random* postavke. Horizontalna osa predstavlja broj korišćenih slučajnih podela (odnosno, broj ko-trening klasifikatora), a vertikalna predstavlja tačnost [%]. U ovom primeru je u toku eksperimenta korišćeno maksimalno 20 različitih slučajnih podela obeležja. Svaka podela rezultuje jednim ko-trening klasifikatorom. Performanse *Best Random* postavke predstavljaju performanse najboljeg od uočenih 20 ko-trening klasifikatora, zbog čega su ove performanse predstavljene referentnom linijom koja se ne menja sa brojem slučajnih podela. Performanse *Random* postavke se interpretiraju na sledeći način: za 5 korišćenih slučajnih podela *Random* postavka predstavlja prosečnu vrednost performansi 5 ko-trening klasifikatora nastalih korišćenjem datih 5 podela. Zbog ovoga performanse *Random* postavke variraju sa brojem korišćenih podela.

Performanse postavki *All*, *L*, *Natural*, $MaxInd_{best}$ i *Best random*, na koje ne utiče broj korišćenih slučajnih podela, predstavljene su ravnim linijama, a performanse *Random*, *MV*, *RSSalg* i $RSSalg_{best}$ podela su predstavljene tačkama. Kao indikator promene vrednosti, za ove podele je prikazan i polinom četvrtog stepena fitovan[16] na dobijene vrednosti.

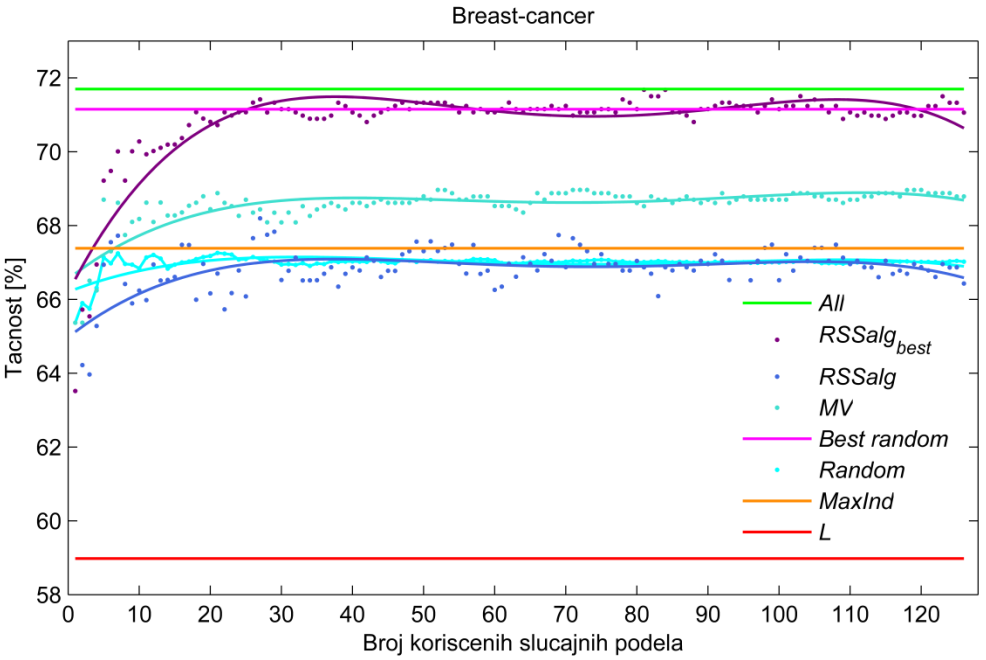


Grafikon 2 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na News2x2 skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.

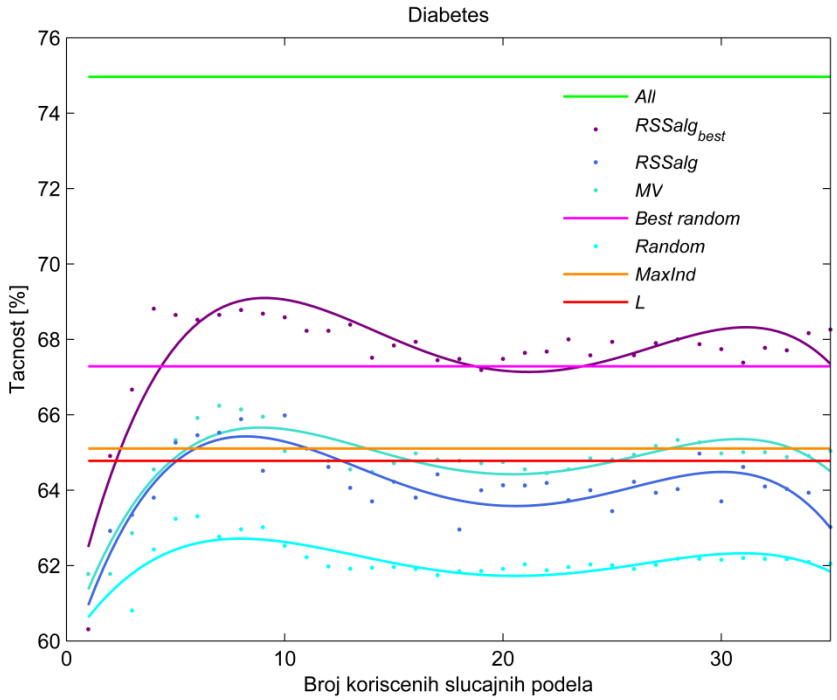
[16] Primenom metode najmanjih kvadrata.



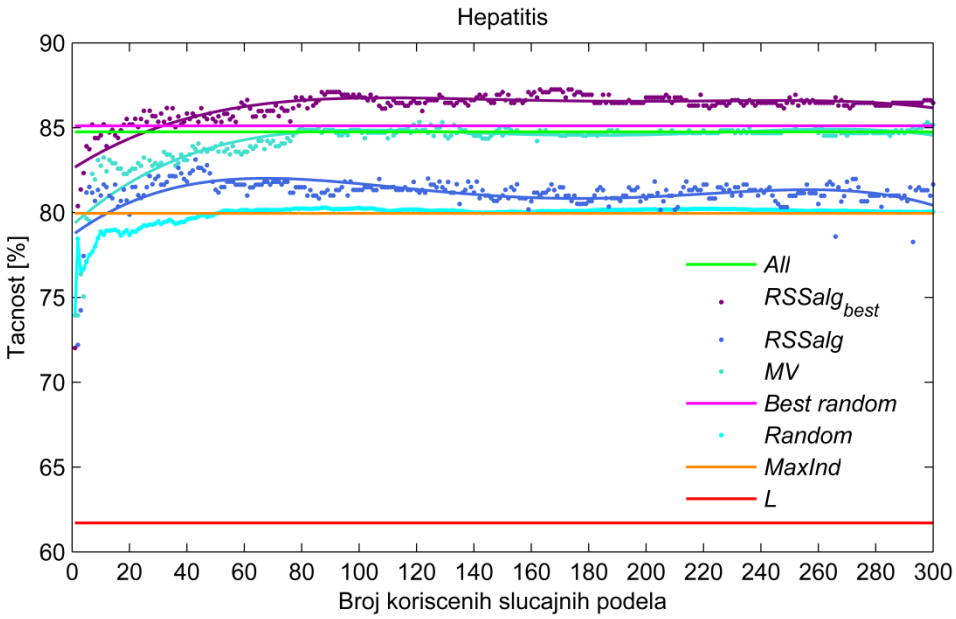
Grafikon 3 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na LingSpam skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.



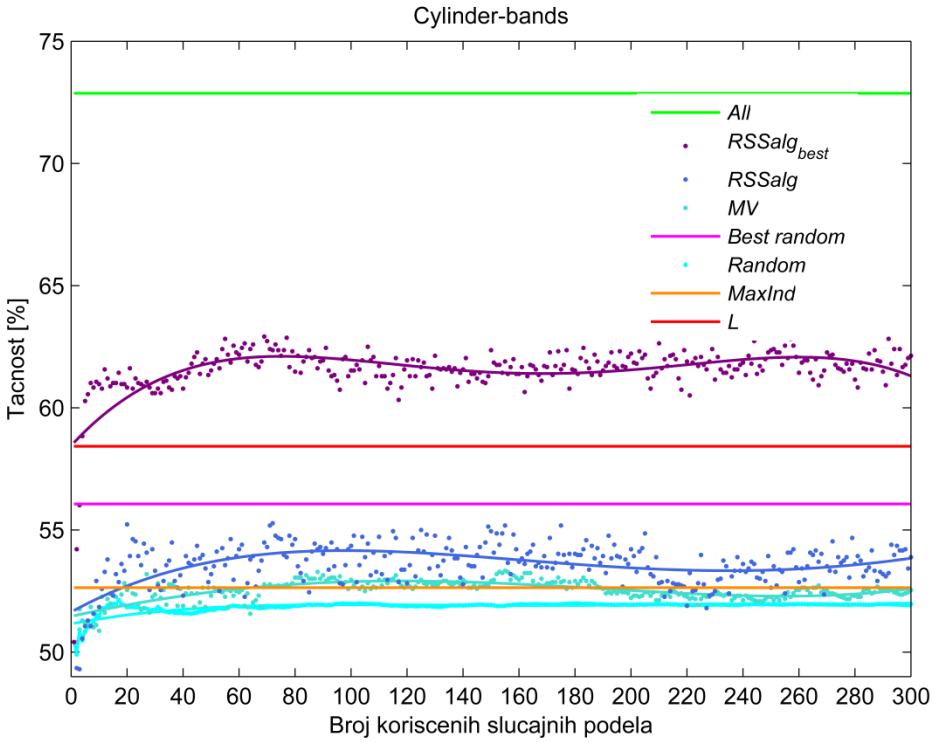
Grafikon 4 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na Breast-cancer skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.



Grafikon 5 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na Diabetes skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.



Grafikon 6 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na Hepatitis skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.



Grafikon 7 Uticaj broja korišćenih slučajnih podela na performanse testiranih postavki na Cylinder-bands skupu podatka. Horizontalna osa predstavlja broj korišćenih slučajnih podela, a vertikalna predstavlja tačnost.

Sa grafikona možemo uočiti sledeće:

1. Postavke *Random*, *MV*, *RSSalg* i $RSSalg_{best}$ su prilično robustne na broj korišćenih podela. Za manji broj podela, performanse ovih algoritama su očekivano manje, ali zatim ravnomerno rastu do određene vrednosti nakon koje ne variraju u velikoj meri. Broj slučajnih podela *m_opt* nakon koga se performanse rešenja dodavanjem novih klasifikatora ne menjaju u značajnoj meri, kao i standardna devijacija izmerene tačnosti algoritama nakon podele *m_opt* je predstavljen u tabeli 8.

Skup podataka	**Preporučeni broj podela *m_opt***	**Standardna devijacija tačnosti algoritma nakon *m_opt* broja podela**			
		Random	*MV*	*RSSalg*	$RSSalg_{best}$
News2x2	100	0.09	0.08	0.50	0.43
LingSpam	60	0.06	0.12	0.53	0.59
Breast-cancer	30	0.03	0.18	0.34	0.17
Diabetes	10	0.16	0.23	0.62	0.34
Hepatitis	80	0.07	0.17	0.46	0.27
Cylinder-bands	80	0.03	0.31	0.68	0.45

Tabela 8 Broj slučajnih podela nakon koga se performanse klasifikatora *MV*, *RSSalg* i $RSSalg_{best}$ prestaju menjati u značajnoj meri.

Dakle, budući da se performanse rešenja ne degradiraju dodavanjem novih klasifikatora, preporučeno je koristiti što je moguće veći broj podela kako bi smo bili sigurni da smo dobili približno maksimalne performanse. Ovde izvršeni eksperimenti daju indikaciju da je na većini skupova podataka dovoljno izvršiti oko 100 podela. Jedini izuzetak je LingSpam skup podataka gde performanse počinju ponovo da rastu oko 250. podele. Na kraju, potrebno je napomenuti da je najmanja standardna devijacija zabeležena kod *Random* postavke. Podsetimo se da *Random* ne predstavlja samo jedan ko-trening klasifikator već predstavlja uprosečenu vrednost performansi svih, do tada izvršenih, ko-treninga sa slučajnom podelom zbog čega je za nju izračunata standardna devijacija mala.

2. Performanse *MV*, *RSSalg* i $RSSalg_{best}$ postavki su korelirane sa performansama *Random* postavke. Ovo je i očekivano budući da su kod tehnika učenja sa grupom hipoteza performanse grupe u korelaciji sa performansama članova grupe [Dietterich 2000].
3. Na Hepatitis, Diabetes i Breast-cancer skupovima podataka performanse $RSSalg_{best}$ su u rangu performansi ko-treninga sa najboljom pronađenom slučajnom podelom (*Best random*). Na preostalim skupovima podataka, performanse $RSSalg_{best}$ postavke su veće. Dakle, u slučaju $RSSalg_{best}$ postavke, kombinovanjem predikcija više klasifikatora se dobijaju bolje ili iste performanse od performansi najboljeg pojedinačnog člana grupe klasifikatora. Za većinsko glasanje (*MV*) to nije uvek slučaj – *MV* postavka je svuda pokazala performanse manje od *Best Random* postavke. *RSSalg* postavka je bolja od *Best Random* postavke na skupovima podataka prirodnog jezika (News2x2 i LingSpam), ali je gora od *Best Random* postavke na UCI skupovima podataka. Potrebno je napomenuti da se *Best random* postavka ne bi mogla koristiti u praksi jer je usled nedostatka anotiranih primera, ovu podelu teško ili čak nemoguće identifikovati[17].

4.5.2 Uticaj odabranih pragova (pojave instance i slaganja anotacije) na performanse *RSSalg* algoritma

U ovom odeljku je testiran uticaj vrednosti pragova pojave instance i slaganja anotacije na performanse *RSSalg* algoritma.

U ovde izvedenom eksperimentu prvo je izgenerisana statistika slučajnih podela za *RSSalg* postavku. Prilikom generisanja statistike za broj slučajnih podela *m* je za svaki od skupova podataka uzeta vrednost navedena u tabeli 8, dok su svi ostali parametri fiksirani na vrednosti izlistane u odeljku 4.2. Nakon generisanja statistike, generisan je niz parova pragova pojave instance i slaganja

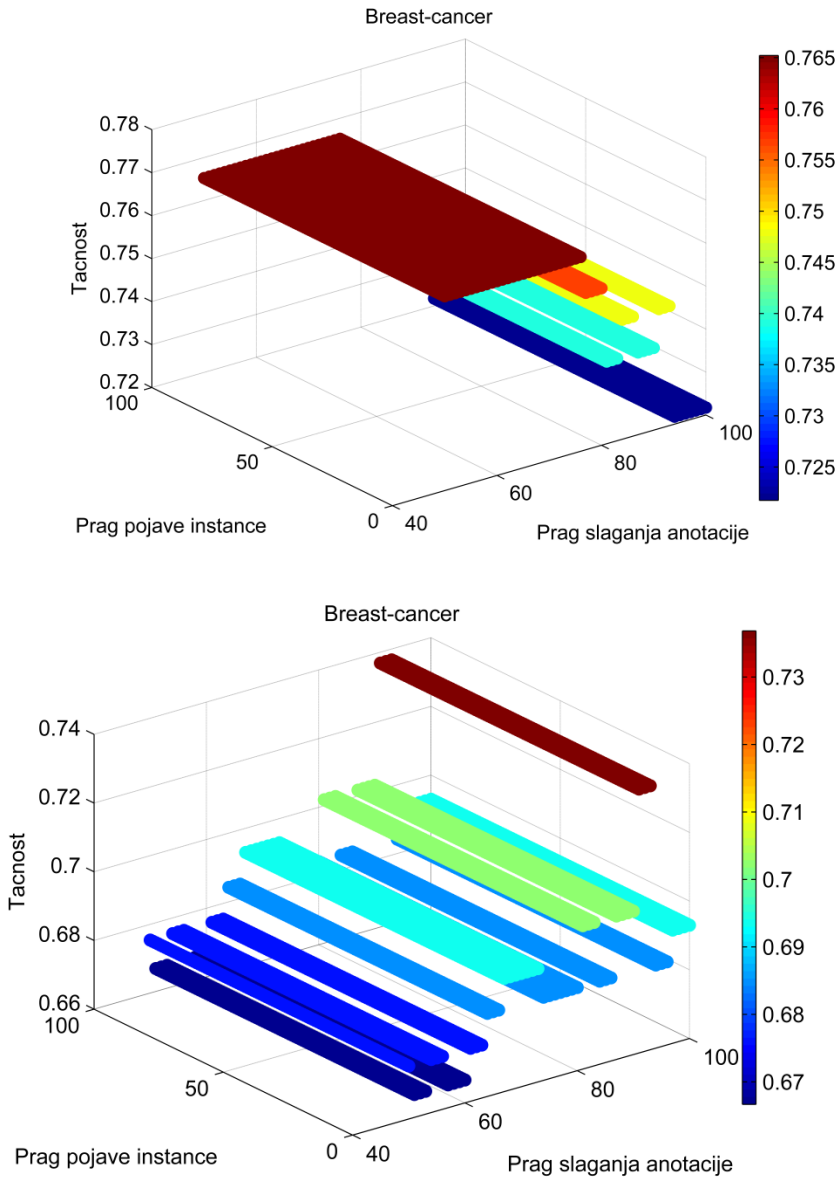
[17] Prema eksperimentima izvedenim u [Du 2010], čak i ako bi smo znali tačno koje karakteristike podele bismo trebali meriti, moguće je da ove karateristike ne bismo mogli evaluirati na malom inicijalnom anotiranom skupu kojim raspolažemo.

anotacije. Prag slaganja anotacije se kreće u opsegu od 50[18]–100 % i menja se sa korakom od 1%. Izračunat je minimalan prag pojave instance u statistici Ets_{min}. Za prag pojave instance su uzete vrednosti iz opsega od 0–100% sa korakom od 1% i te vrednosti su skalirane na opseg Ets_{min}–100%[19]. Za ovako dobijene vrednosti pragova su generisane sve moguće kombinacije. Pomoću svakog generisanog para pragova iz zabeležene statistike su eliminisane instance koje prema zadatim pragovima nisu pouzdano anotirane. Nakon toga je na preostalim instancama nadgledanim obučavanjem treniran finalni klasifikator (NB model) i izmerena je tačnost modela postignuta na test skupu. Takođe je za iste vrednosti pragova izmerena i estimacija tačnosti finalnog modela prema metodi predloženoj u odeljku 3.2.1.

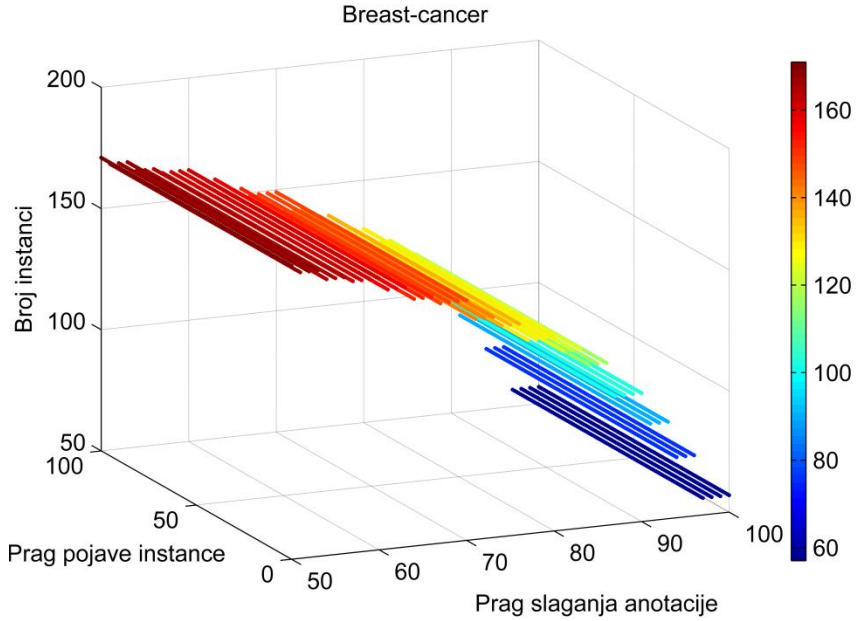
U ovom eksperimentu za evaluaciju nije korišćena procedura unakrsne validacije opisana u poglavlju 4, već je izvršena samo jedna podela na anotirani, neanotirani i test skup (korišćena je samo jedna podela od 10 dobijenih unakrsnom validacijom). Razlog za ovo jeste što je utvrđeno da $RSSalg_{best}$ postavka za različite podele na anotirani, neanotirani i test skup (u različitim krugovima unakrsne validacije) rezultuje veoma različitim pragovima. Forsiranjem da se za sve krugove unakrsne validacije koriste isti pragovi i uprosečavanjem dobijenih performansi se dobijaju znatno lošije performanse *RSSalg* postavke (pragovi idealni za jednu podelu mogu biti izuzetno loši za drugu) i tim uprosečavanjem je teško pratiti realnu zavisnost ciljne funkcije koja se optimizuje (tačnost finalnog klasifikatora) od vrednosti odabranih pragova. Pojava da za isti skup podataka dve različite podele na anotirani, neanotirani i test skup rezultuju veoma različitim zaključcima u pogledu izbora pragova je demonstrirana na Breast-cancer skupu podataka za koga su prikazani rezultati dve različite podele na anotirani, neanotirani i test skup, pri čemu su sve ostale korišćene vrednosti parametara iste. Na grafikonu 8 predstavljen je uticaj vrednosti pragova na tačnost finalnog modela za prvu (gornji grafikon) i drugu (donji grafikon) podelu na anotirani, neanotirani i test skup izvršenu na Breast-cancer skupu podataka. Na grafikonu 9 je predstavljena zavisnost broja instanci zadržanih u skupu L_{int} (broj instanci za koje se smatra da su pouzdano anotirane) od odabranih vrednosti pragova za prvu podelu. Isti grafik za drugu podelu je izostavljen budući da je veoma sličan grafikonu 9.

[18] Budući da se ovde eksperimentiše na binarnim klasifikacionim problemima, najgora vrednost koja se može dobiti za slaganje anotacije je 50% (polovina klasifikatora svrstava instancu u jednu klasu, a druga polovina je svrstava u drugu klasu). Ovo nije slučaj za višekategorijske probleme gde neslaganje klasifikatora može biti i veće.

[19] Ne vredi ispitivati pragove pojave instance ispod Ets_{min} (odeljak 3.2.1). Zato se u ovom eksperimentu oseg Ets_{min} – 100% tretira kao da se radi o opsegu 0-100%. Skaliranje praga p se radi prema formuli $Ets_{min} + p \cdot (100 - Ets_{min})/100$. Na primer, neka je instanca koja se pojavljuje u najmanje uvećanih anotiranih skupova prisutna u 30% skupova. U ovom slučaju se prag 0% skalira na 30%, a npr. prag 50% se skalira na 65% (sredina opsega 30-100%).



Grafikon 8 Uticaj vrednosti odabranih pragova na performanse *RSSalg* postavke za dve različite podele na anotirani, neanotirani i test skup na **Breast-cancer** skupu podataka**.** ***X*-osa** predstavlja prag slaganja anotacije. ***Y*-osa** predstavlja prag pojave instance**.** ***Z*-osa** predstavlja postignutu tačnost *RSSalg* postavke. Postignuta tačnost je kodirana i bojama, a toplotna mapa ovih boja je predstavljena na desnoj strani slike. **Gore** je za **prvu podelu na anotirani, neanotiran i test skup** predstavljen uticaj vrednosti pragova na performanse *RSSalg* postavke. Tačnost NB modela obučenog na inicijalnom skupu L za ovu podelu iznosi L_{acc} = 0.626, a tačnost NB modela obučenog na skupu *All* iznosi All_{acc} = 0.730. **Dole** je za **drugu podelu na anotirani, neanotirani i test skup** predstavljen uticaj vrednosti pragova na performanse *RSSalg* postavke. Tačnost NB modela obučenog na inicijalnom skupu L za ovu podelu iznosi L_{acc} = 0.491, a tačnost NB modela obučenog na skupu *All* iznosi All_{acc} = 0.711.



Grafikon 9 Uticaj vrednosti odabranih pragova na broj instanci zadržanih u skupu L_{int} prilikom formiranja obučavajućeg skupa za finalni klasifikator u *RSSalg* postavci. Radi se o **prvoj podeli na anotirani, neanotirani i test skup** na skupu podataka **Breast-cancer. *X*-osa** predstavlja prag slaganja anotacije. ***Y*-osa** predstavlja prag pojave instance. **Z-osa** predstavlja broj instanci uključen u obuku finalnog modela za različite vrednosti pragova. Broj zadržanih instanci je kodiran i bojama, a toplotna mapa ovih boja je predstavljena na desnoj strani slike.

Sa grafikona 8 možemo zaključiti da na Breast-cancer skupu podataka prag pojave instance nema uticaja na tačnost formiranog modela. Sa grafikona 9 se vidi da prag pojave instance nema uticaja ni na broj selektovanih instanci (gotovo sve instance L_{int} skupa su jednako zastupljene), tako da je to razlog zbog koga ovaj prag ne utiče na tačnost finalnog modela[20]. Za prvu podelu najveća postignuta tačnost je dobijena za opseg praga slaganja anotacije 50–76%, dakle kada su sve ili barem većina instanci zadržane u statistici. Ova tačnost iznosi 0.765, što je u ovom slučaju veće od All_{acc} tačnosti od 0.730. Za drugu podelu najveća postignuta tačnost je dobijena za opseg praga slaganja anotacije 91–93%, dakle kada je većina instanci isključena iz statistike. Ova tačnost iznosi 0.737, što je veće od All_{acc} tačnosti od 0.711 za ovu postavku. Odavde vidimo da, čak i na istom skupu podataka, uz iste vrednosti ostalih parametara, izbor polaznog anotiranog, neanotiranog i test skupa u značanoj meri utiče na ciljnu optimizacionu funkciju. Potrebno je još istaći da nijedan od testiranih pragova (za obe podele) u ovom slučaju nije doveo do degradacije performansi polaznog klasifikatora. Čak i uz loš izbor praga slaganja anotacije, minimalna izmerena

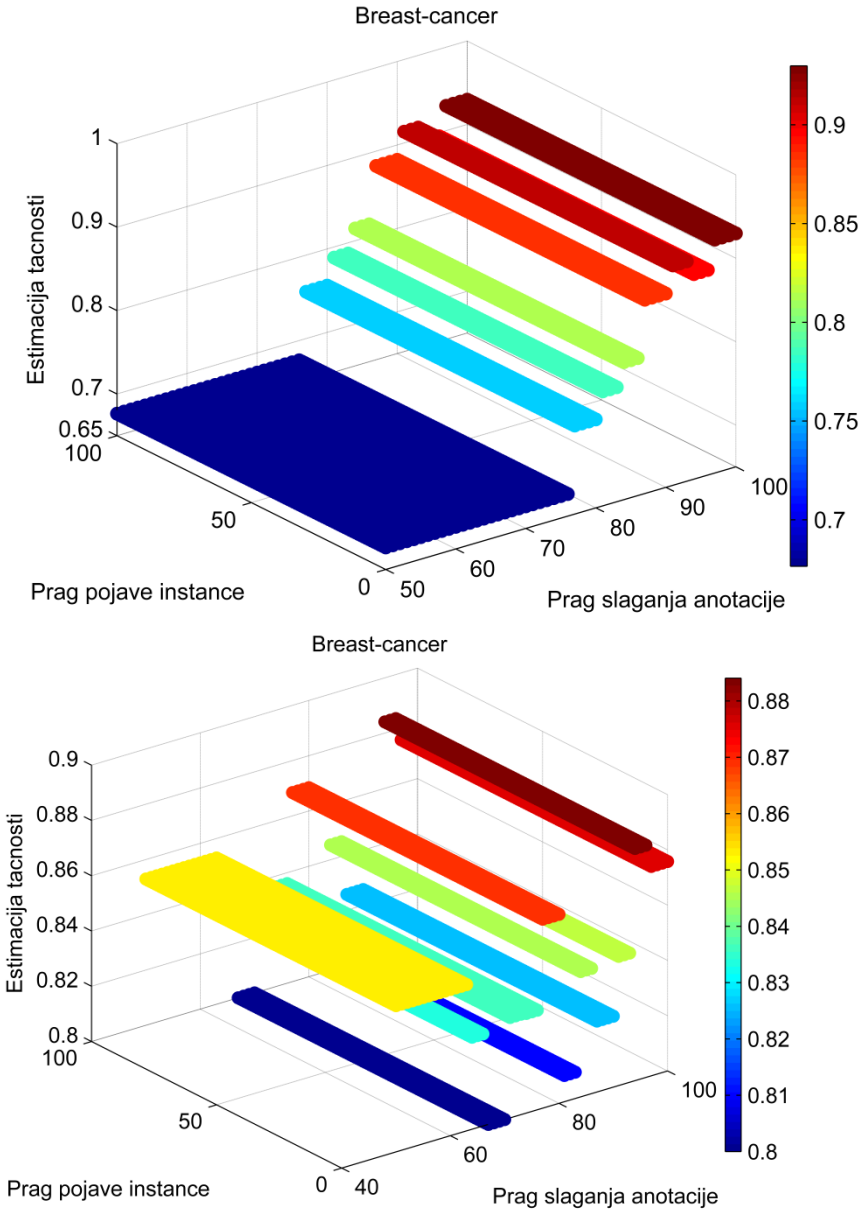
[20] Razlog zbog koga se sve instance nalaze u svim skupovima leži u postavci eksperimenta. Breast-cancer je veoma mali skup podataka, tako da je ko-trening procesom u izvršenih 20 iteracija izvršena anotacija svih instanci neanotiranog skupa podataka.

tačnost za prvu podelu iznosi 0.722, što je značajno veće od polazne tačnosti L_{acc} = 0.626 za ovu podelu, a minimalna tačnost za drugu podelu iznosi 0.667, što je takođe značajno veće od polazne tačnosti L_{acc} = 0.491 za ovu podelu.

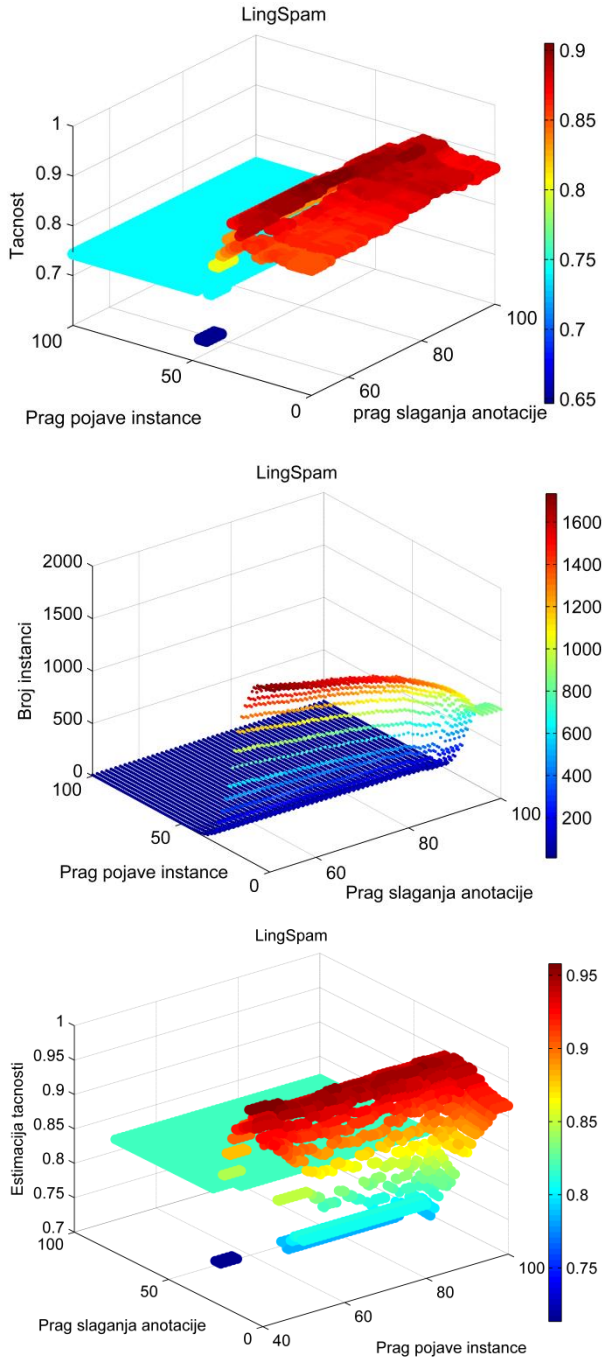
Međutim, iz ovih rezultata se takođe vidi da su performanse *RSSalg* postavke prilično osetljive na izbor pragova. Za prvu podelu izmerena tačnost varira između 0.722 i 0.765 (razlika tačnosti od 4.3%), a tačnost za drugu podelu varira između 0.667 i 0.737 (razlika tačnosti od 7%). Postavlja se pitanje – koliko dobro možemo estimirati tačnost modela za zadate pragove primenom postupka opisanog u odeljku 3.2.1? Za prvu podelu na Breast-cancer skupu podataka očekujemo relativno lošu estimaciju tačnosti pošto najbolji model rezultuje zadržavanjem najvećeg broja instanci statistike za obuku modela, a ovo ostavlja previše mali skup statistike za evaluaciju. Za drugu podelu, iz istog razloga očekujemo bolje rezultate. Grafički prikaz estimacije tačnosti za opisane podele na Breast-cancer skupu podataka se nalazi na grafikonu 10. Za prvu podelu na Breast-cancer skupu podataka je predloženi način estimacije tačnosti zaista omanuo i vratio rezultat da se najbolji opseg tačnosti nalazi između 97 i 100%, što rezultuje realnom tačnošću od 0.722, što predstavlja najgore moguće rešenje. Za drugu podelu estimacija tačnosti je dovela do zaključka da se najveća tačnost modela dobija za opseg praga slaganja anotacije od 94 – 96% što rezultuje realnom tačnošću modela od 0.684, što je veće od minimalne tačnosti kojom je estimacija mogla rezulovati (0.667), ali nije dovelo do ciljne tačnosti od 0.737.

Drugačiji primer, takođe uočen na LingSpam skupu podataka, je predstavljen na grafikonu 11. Prvi grafik (gledano od gore) na grafikonu 11 predstavlja zavisnost tačnosti modela od izabranih vrednosti pragova. Vidi se da u ovom slučaju na tačnost finalnog modela znatno veći uticaj vrši prag pojave instance od praga slaganja anotacije (iako oba praga imaju uticaj). Ako pogledamo poslednji grafik (gledano odgore) na na grafikonu 11 koji predstavlja zavisnost broja instanci zadržanih za obuku finalnog modela od izabranih vrednosti pragova, vidimo da za anotirane instance postoji prilično veliko slaganje po pitanju dodeljene anotacije (instance se eliminišu iz skupa više po pitanju praga pojave instance nego praga slaganja anotacije). Najveća tačnost modela je izmerena za opseg praga anotacije od 67–73% i praga pojave instance od 28–30% i iznosi 0.905, što je u ovom slučaju veće od Acc_{all} tačnosti od 0.865. Najmanja izmerena tačnost *RSSalg* postavke iznosi 0.647, što je manje od tačnosti polaznog modela od L_{acc} = 0.744. I ovde se vidi da su performanse *RSSalg* postavke veoma osetljive na odabrane vrednosti pragova: izmerena razlika u tačnosti za najbolji i najgori odabir pragova iznosi čak 25.8%. Srednji grafik na grafikonu 11 predstavlja estimaciju tačnosti modela izračunatu postupkom opisanim u odeljku 3.2.1. Vidimo da je u datom slučaju estimacija tačnosti prilično dobra – dovela je do zaključka da se najveća tačnost modela dobija za opseg praga slaganja anotacije od 50–54% i opseg praga pojave

instance od 28–29% što rezultuje realnom tačnošću modela od 0.888, što je relativno blisko maksimalnoj mogućoj tačnosti od 0.905.



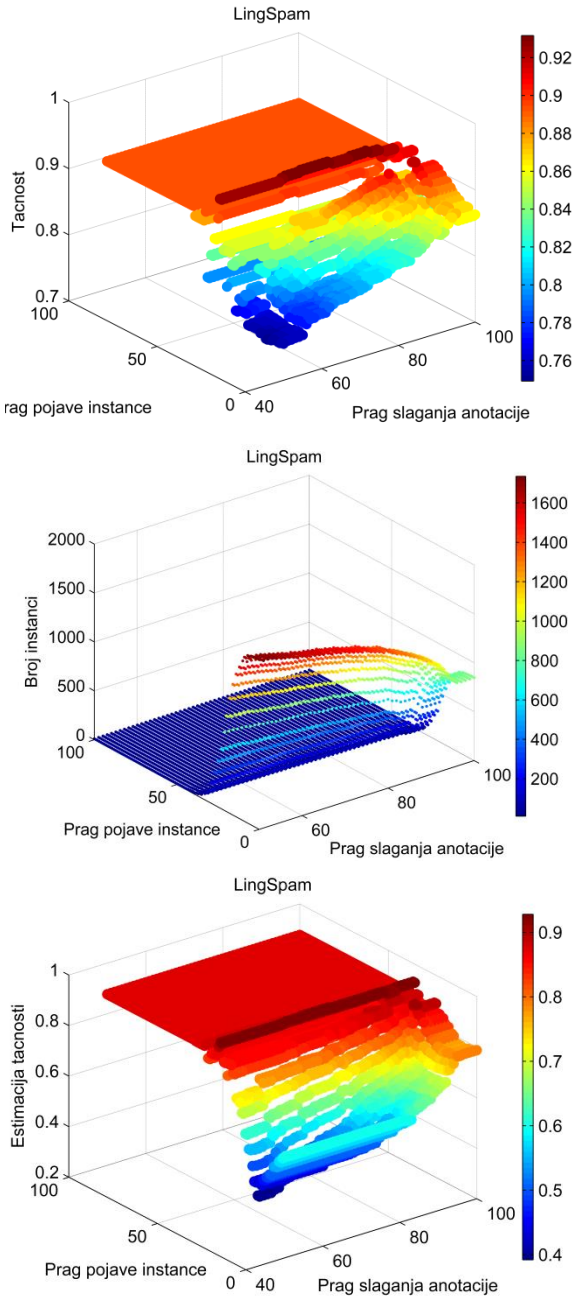
Grafikon 10 Estimacija tačnosti za vrednosti odabranih pragova na **Breast-cancer** skupu podataka**.** ***X*-osa** predstavlja prag slaganja anotacije. ***Y*-osa** predstavlja prag pojave instance. ***Z*-osa** predstavlja estimiranu tačnost *RSSalg* postavke. Izmerena estimacija tačnosti je kodirana i bojama, a toplotna mapa ovih boja je predstavljena na desnoj strani slike. **Gore** je za **prvu podelu na anotirani, neanotiran i test skup (grafikon 8, gore)** predstavljena estimacija tačnosti za zadate vrednosti pragova. **Dole** je za **drugu podelu na anotirani, neanotirani i test skup (grafikon 8, dole)** predstavljena estimacija tačnosti za zadate vrednosti pragova.



Grafikon 11 Uticaj vrednosti odabranih pragova na tačnost *RSSalg* postavke (gore), broj instanci zadržan za obučavanje finalnog modela (sredina) i na vrednost estimacije tačnosti *RSSalg* postavke (dole) za slučajnu podelu na anotirani, neanotirani i test skup na **LingSpam** skupu podataka. ***X*-osa** predstavlja prag slaganja anotacije. ***Y*-osa** predstavlja prag pojave instance. Tačnost NB modela obučenog na inicijalnom skupu L za ovu podelu iznosi L_{acc} = 0.744, a tačnost NB modela obučenog na skupu *All* za ovu podelu iznosi All_{acc} = 0.865.

Na grafikonu 12 je istaknut slučaj, uočen na LingSpam skupu podataka, gde oba praga imaju veliki uticaj na rezultujuću tačnost modela. Performanse polaznog modela su u zadatom slučaju već veoma visoke: L_{acc} = 0.894. Tačnost modela koji se dobija uključivanjem dodatnih anotiranih primera je manja od polazne[21]: All_{acc} iznosi 0.852. Ranije studije pokazuju da kada da su performanse polaznog modela već veoma visoke, ko-trening nema prostora da unapredi polazne performanse [Pierce 2001; Mihalcea 2004]. Međutim, najveća tačnost izmerena za *RSSalg* postavku iznosi 0.932 za opseg praga slaganja anotacije od 71 do 78% i opseg praga pojave instance od 35 do 36%. Ovaj model ima više koristi od uključivanja manjeg broja instanci anotiranih od strane većeg broja klasifikatora i kod kojih je slaganje anotacije veliko. Najgora tačnost *RSSalg* postavke se dobija za prag slaganja anotacije od oko 50% i prag pojave instance od oko 10% i iznosi 0.749. Dakle, vidimo da su i ovde performanse *RSSalg* postavke veoma zavisne od odabranih vrednosti parametara: razlika maksimalne i minimalne izmerene tačnosti iznosi 18.3%. U ovom slučaju je estimacija tačnosti pronašla prilično dobar model: prema estimaciji, najbolji izbor za prag slaganja anotacije je 71–78%, a za prag pojave instance 35–36%, a realna tačnost modela u ovom slučaju iznosi 0.932 – dostignuta je maksimalna moguća tačnost.

[21] LingSpam skup podataka je relativno neizbalansiran po pitanju klase: tačnost koja se dobija kada se sve instance svrstaju u većinsku klasu iznosi 0.843. Zbog ovoga je proverena i *f*-mera klasifikatora treniranih na skupovima *L* i *All*, međutim *f*-mera *All* klasifikatora je takođe manja od *f*-mere *L* klasifikatora za obe klase prisutne u LingSpam skupu podataka. Navedeni efekat najverovatnije proizilazi iz šuma u podacima ili iz slučajno povoljno odabranih polaznih i test podataka.



Grafikon 12 Uticaj vrednosti odabranih pragova na tačnost *RSSalg* postavke (gore), broj instanci zadržan za obučavanje finalnog modela (sredina) i na vrednost estimacije tačnosti *RSSalg* postavke (dole) za slučajnu podelu na anotirani, neanotirani i test skup na **LingSpam** skupu podataka. ***X*-osa** predstavlja prag slaganja anotacije. ***Y*-osa** predstavlja prag pojave instance. ***Z*-osa** predstavlja postignutu tačnost *RSSalg* postavke. Postignuta tačnost je kodirana i bojama, a toplotna mapa ovih boja je predstavljena na desnoj strani slike. Tačnost NB modela obučenog na inicijalnom skupu L je za ovu podelu $L_{acc} = 0.894$, a tačnost NB modela obučenog na skupu *All* je $All_{\text{acc}} = 0.852$.

U ovom odeljku je prikazano nekoliko karakterističnih primera zavisnosti tačnosti *RSSalg* klasifikatora od odabranih parametara za selekciju pouzdano anotiranih instanci. Na osnovu njih je zaključeno sledeće:

- Prilikom selekcije instanci oba praga (prag pojave instance i prag slaganja anotacije) imaju uticaja na tačnost finalnog modela i zbog toga ima smisla koristiti oba ova praga.
- Performanse *RSSalg* postavke su veoma osetljive na izbor pragova. Pokazano je da pogrešan odabir pragova može uzrokovati da ovaj postupak ne unapredi polazni model ili da čak degradira performanse polaznog modela.
- Nije utvrđena generalna preporuka za izbor pragova. Optimizaciona funkcija je veoma osetljiva na izbor polaznog anotiranog, neanotiranog i test skupa. Čak i na istom skupu podataka, za dve različite podele mogu da važe različita pravila: za jednu podelu može biti bolje da se u proces obuke uključi što više instanci, bez obzira na pragove, dok za drugu podelu može biti pogodnije da se instance selektuju restriktivno. Zbog toga vredi zadržati GA kao optimizacioni model kojim se mogu pronaći rešenja ovako kompleksnog problema pretrage.
- Estimacija tačnosti modela koji rezultuje zadatim pragovima nije pouzdana na svim skupovima podataka. U budućnosti bi trebalo eksperimentisati sa alternativnim mogućnostima procene tačnosti modela. Jedna mogućnost jeste primena unakrsne validacije, međutim ovo je vremenski prilično skupa procedura. Druga opcija koja ostaje da se istraži jeste mogućnost povezivanja sa mehanizmima editovanja podataka (*data editing mechanisms*) u cilju identifikacije pogrešno anotiranih instanci [Muhlenbach 2004], kao što je predloženo u [Zhou 2005].

U ovom odeljku su, radi preglednosti, istaknuti reprezentativni primeri uticaja pragova na performanse *RSSalg* postavke koji se odnose na nekoliko različitih slučajnih podela na anotirane/neanotirane i test podatke na dva skupa podataka (Breast-Cancer i LingSpam). Potrebno je napomenuti da se ovde navedeni zaključci generalizuju i na ostale skupove podataka na kojima je testiran uticaj pragova na performanse *RSSalg* postavke (News2x2, Diabetes, Hepatitis i Cylinder-bands).

4.5.3 Uticaj redudantnosti skupa podataka

Već je obrazloženo da se bolje performanse *Random* postavke, kao i svih postavki baziranih na njoj (*MV*, *RSSalg*, $RSSalg_{best}$ i *IMCC*) očekuju na redudantnijim skupovima podataka. Zaista, ako posmatramo grafike performansi predstavljene u odeljku 4.5.1, vidimo da, uz dovoljan broj slučajnih podela i $RSSalg_{best}$ i *RSSalg* postavka postižu bolje performanse od ostalih postavki na redudantnijim skupovima podataka News2x2 i LingSpam.

U eksperimentima izvedenim u odeljku 4.5.1 $RSSalg_{best}$ postavka je u rangu ili čak i nadmašuje performanse *All* postavke na News2x2, LingSpam,

Hepatitis i Breast-cancer skupovima podataka, dok je na Diabetes i Cylinder-bands skupovima podataka u značajnoj meri gora od *All* postavke. Lošije performanse $RSSalg_{best}$ postavke na Diabetes skupu podataka bi se mogle objasniti malim brojem obeležja. Usled malog broja obeležja (svega 8), ovde je bilo moguće izvesti samo 35 slučajnih podela na dva pogleda jednakih veličina. Takođe, moguće da na osnovu malog broja obeležja u posebnim klasifikatorima nije moguće dovoljno dobro izraziti ciljnu funkciju. Međutim, skup podataka Cylinder-bands ima 39 obeležja, što je u značajnoj meri više od broja obeležja Breast-cancer skupa podataka na kome se $RSSalg_{best}$ postavka pokazala kao uspešna. Da bi smo ispitali da li je redudantnost skupa podataka (odnosno njeno odsustvo) razlog što $RSSalg_{best}$ ima loše performanse na Cylinder-bands skupu podataka, potrebno je da je na neki način kvantifikujemo.

U radu [Salaheldin 2010] je predložena veštačka podela obeležja za ko-trening koja se bazira na istovremenoj maksimizaciji snage pojedinačnih klasifikatora i maksimizaciji različitosti (*diversity*) između njih. Funkcija čijom se minimizacijom dobija opisana podela je:

$$maxDiv = E(V_1) + E(V_2) + \frac{E(V_1) + E(V_2)}{E(V_1 * V_2)}, \tag{62}$$

gde $E(V_i)$ predstavlja entropiju *i*-tog pogleda, a $E(V_1 * V_2)$ predstavlja entropiju kombinovanog klasifikatora. Entropija klasifikatora izračunata na skupu podatka *X* se definiše na sledeći način:

$$H(X) = -\sum_{i=1}^{n} p(x_i) log_b(p(x_i)) \tag{63}$$

gde *n* predstavlja broj instanci skupa *X*, $p(x_i)$ predstavlja verovatnoću da klasifikator tačno klasifikuje instancu x_i, a *b* je korišćena baza logaritma. Entropijom u ovom slučaju modelujemo nesigurnosti klasifikatora – veća entropija znači manju pouzdanost u klasifikator. U [Salaheldin 2010] je istaknuto da bi idealna mera jačine pogleda bila njegova tačnost izmerena na test skupu. Međutim, budući da u ko-trening postavci labele neanotiranih instanci nisu dostupne, tačnost klasifikatora na celom skupu je nemoguće meriti. Zbog toga se ona procenjuje na osnovu predložene mere entropije.

Da bismo videli koliko redudantnost korelira sa uspehom *RSSalg* postavke, usvojićemo ovu meru za procenu kvaliteta pojedinačnih ko-trening klasifikatora nastalih korišćenjem slučajnih podela obeležja. Za početak, izvršićemo eksperiment gde ćemo za svaki od skupova podataka izračunati vrednost sledeće mere:

$$Div\_st = Acc(V_1) + Acc(V_2) + \frac{Acc(V_1 * V_2)}{Acc(V_1) + Acc(V_2)}, \tag{64}$$

gde $Acc(V_i)$ predstavlja tačnost *i*-tog pogleda izmerenu na test skupu, a $Acc(V_1 * V_2)$ predstavlja tačnost kombinovanog pogleda. Ova mera raste sa porastom tačnosti individualnih klasifikatora, kao i sa porastom jačine kombinovanog klasifikatora naspram jačine njegovih pojedinačnih članova. Kasnije ćemo izvesti i eksperiment u kome ćemo koristiti meru predloženu u [Salaheldin 2010] da vidimo koliko dobro se ovako izračunata mera redudantnosti može proceniti u slučaju nedostatka anotiranih podataka.

Pristup predložen u radu [Salaheldin 2010] je baziran na konceptu predloženom u [Opitz 1999]. U [Opitz 1999] je predložen genetski algoritam za kreiranje grupe klasifikatora. Genetskim algoritmom se maksimizuje jačina individualnih klasifikatora u grupi, kao i njihova međusobna različitost. Svakom novom klasifikatoru *i* koji se dodaje u grupu se dodeljuje funkcija prilagođenosti koja se računa na sledeći način:

$$Fitnes_i = Accuracy_i + \mu * Diversity_i, \quad (65)$$

gde $Accuracy_i$ predstavlja tačnost datog klasifikatora, a *Diversity* modeluje njegovu različitost od grupe klasifikatora i računa se kao razlika tačnosti individualnog klasifikatora i tačnosti kombinovane predikcije grupe klasifikatora. Konstanta μ modeluje kompromis između uticaja tačnosti i uticaja različitosti. Vrednost ove konstante nije baš jednostavno proceniti. U [Opitz 1999] se predlaže automatski način podešavanja ove vrednosti, baziran na grešci populacije i prosečnoj raznolikosti u grupi klasifikatora. Bazirano na [Opitz 1999], za svaki skup podatka ovde je izračunata i mera:

$$Div = Acc(V_1 * V_2 * ... * V_K) - \frac{\sum_{k=1}^{K} Acc(V_k)}{K}, \quad (66)$$

gde $Acc(V_i)$ predstavlja tačnost *i*-tog pogleda, $Acc(V_1 * V_2 * ... * V_k)$ predstavlja predstavlja tačnost kombinovanog pogleda, a *K* predstavlja broj klasifikatora u grupi (u slučaju pojedinačnog ko-trening klasifikatora $K = 2$). Ova mera, za razliku od jednačine (64), gde se istovremeno procenjuje i tačnost i različitost klasifikatora, meri isključivo različitost klasifikatora.

Eksperiment je izveden primenom 100 različitih slučajnih podela na svakom od skupova podataka[22], pri čemu je merena prosečna tačnost nastalih individualnih klasifikatora i prosečna tačnost kombinacije ovih klasifikatora. Individualni klasifikatori su obučavani na skupu podataka nastalog spajanjem anotiranog i neanotiranog skupa. Rezultati ovog eksperimenta su prikazani u tabeli 9. Skupovi podataka u ovoj tabeli su raspoređeni po opadajućoj vrednosti *Div_st* mere (jednačina 64).

[22] Osim u slučaju Diabetes skupa podataka, gde je korišćen maksimalan broj od 35 podela.

Skup podataka	$Acc(V_1)$	$Acc(V_2)$	$Acc(V_1{*}V_2)$	$avg(Acc)$	*Div*	*Div_st*
News2x2	0.864	0.866	0.897	0.865	0.032	2.25
LingSpam	0.856	0.859	0.894	0.858	0.037	2.24
Hepatitis	0.815	0.825	0.846	0.82	0.026	2.16
Breast-cancer	0.709	0.712	0.723	0.711	0.013	1.93
Diabetes	0.708	0.717	0.753	0.713	0.041	1.95
Cylinder-bands	0.676	0.683	0.720	0.68	0.041	1.89

Tabela 9 Prosečna jačina i različitost pojedinačnih klasifikatora nastalih primenom slučajnih podela obeležja, obučenih na skupu nastalom spajanjem anotiranog i neanotiranog skupa podataka. Notacija: $Acc(V_1)$: prosečna tačnost prvog klasifikatora na test skupu; $Acc(V_2)$: prosečna tačnost drugog klasifikatora na test skupu; $Acc(V_1{*}\ V_2)$: prosečna tačnost kombinovanog klasifikatora na test skupu; *avg*(*Acc*): prosek tačnosti $Acc(V_1)$ i $Acc(V_2)$; *Div*: različitost izračunata prema formuli (66); *Div_st*: kombinacija jačine i različitosti klasifikatora izračunata prema formuli (64).

Radi veće preglednosti, ovde je u tabeli 10 ponovljen deo rezultata predstavljenih u tabeli 2 uz vrednosti *avgAcc, Div* i *Div_st* mera predstavljenih u u tabeli 9 za odgovarajuće skupove podataka, a skupovi podataka su takođe poređani po opadajućoj vrednosti dobijene *Div_st* mere.

Datasets	RSSalg	RSSalg best	IMCC	$Acc(V_1{*}V_2)$	*Div*	*Div_st*
News2x2	89.1±3.1	**90.6±1.8**	85.7±0.7	0.897	0.032	2.25
LingSpam	88.5±7.0	91.1±5.9	**95.3±0.5**	0.894	0.037	2.24
Hepatitis	82.6±3.5	**86.5±3.4**	85.8±5.1	0.846	0.026	2.16
Breast-cancer	67.0±6.8	70.4±5.3	**73.9±2.2**	0.723	0.013	1.93
Diabetes	63.9±3.7	67.7±1.8	**71.7±2.1**	0.753	0.041	1.95
Cylinder-bands	54.3±4.8	61.6±2.5	**65.9±1.9**	0.720	0.041	1.89

Tabela 10 Prve tri kolone (gledano sa leva) predstavljaju deo tabele 2 iz odeljka 4.4 ponovljen radi preglednosti rezultata: poređenje alternativnih ko-trening postavki. U datim kolonama je prikazana postignuta tačnost i standardna devijacija (u procentima) dobijena u proceduri stratifikovane 10-struke unakrsne validacije na svakom od skupova podataka. Tri poslednje kolone (gledano sa leva) predstavljaju vrednosti predstavljene u tabeli 9 iz ovog odeljka: $Acc(V_1{*}\ V_2)$: prosečna tačnost kombinovanog klasifikatora na test skupu; *Div*: različitost izračunata prema formuli (66); *Div_st*: kombinacija jačine i različitosti klasifikatora izračunata prema formuli (64).

Na osnovu tabele 10 možemo zaključiti da i redudantnost i jačina pojedinačnih pogleda u velikoj meri utiču na performanse rešenja. Dva skupa podataka gde $RSSalg_{best}$ nije u stanju da dostigne performanse *All* postavke (Diabetes i Cylinder-bands, odeljak 4.5.1, grafikoni 5 i 7) su zaista poslednja prema *Div_st* meri. Međutim, prema *Div* meri vidi se da je na ova dva skupa podataka u proseku raznolikost pojedinačnih klasifikatora zapravo veća nego na ostalim skupovima podataka. Na osnovu ovoga možemo zaključiti da razlog što $RSSalg_{best}$ postavka na ovim skupovima podataka nema toliko dobre performanse kao na ostalim najverovatnije leži u generalno manjoj tačnosti pojedinačnih klasifikatora (kolona $Acc(V_1{*}V_2)$[23], tabela 10) što rezultuje

[23] Kombinovani klasifikator $Acc(V_1{*}V_2)$ predstavlja jedan ko-trening klasifikator (koji u okviru sebe kombinuje dva klasifikatora V_1 i V_2 bazirana na zasebnim pogledima), a grupa

umanjenom tačnošću grupe klasifikatora. Prikazani rezultati ukazuju na to da *Div_st* mera prilično dobro oslikava odnos tačnosti individualnih klasifikatora i njihove međusobne raznolikosti koji utiče na performanse $RSSalg_{best}$ postavke. Jedini izuzetak je Breast-cancer skup podataka za koji je prema *Div_st* meri procenjeno da je lošiji za primenu ove postavke od Diabetes skupa podataka, što nije slučaj.

Rezultati (grafikoni 2 – 7 u odeljku 4.5.1 i tabela 10) ukazuju da je *RSSalg* postavka još osetljivija na kombinovani uticaj redudantnosti i snage pojedinačnih klasifikatora. Na News2x2 skupu podataka ova postavka je u rangu performansi $RSSalg_{best}$ i *All* postavke i nadmašuje performanse ostalih postavki (grafikon 2). Na LingSpam skupu podataka i dalje nadmašuje tačnost ostalih postavki, ali ne postiže performanse $RSSalg_{best}$ postavke (grafikon 3). Dalje, na Hepatitis skupu podataka ova postavka je razumno dobra – unapređuje performanse polaznog klasifikatora i donekle je jača od $MaxInd_{best}$ i *Random* postavke, ali gubi u značajnoj meri i od $RSSalg_{best}$ i od *MV* postavke (grafikon 6). Na Breast-cancer se izjednačuje sa $MaxInd_{best}$ i *Random* postavkom (grafikon 4), dok na preostalim skupovima podataka – Diabetes i Cylinder-bands čak degradira performanse polaznog klasifikatora (grafikoni 5 i 7, respektivno).

Na osnovu rezultata prikazanih u tabeli 10, sledi da je *IMCC* postavka nešto manje osetljiva na redudantnost od $RSSalg_{best}$ postavke. Vidi se da, sa izuzetkom LingSpam skupa podataka, $RSSalg_{best}$ gubi od *IMCC* postavke upravo na skupovima podataka koji se odlikuju manjim prosekom *Div_st* mere.

Postavlja se pitanje da li je na osnovu podataka raspoloživih u ko-trening postavci moguće proceniti da li je skup podataka dovoljno redudantan za primenu *RSSalg* postavke. U ko-trening postavci nam nedostaju anotirane instance pomoću kojih bi smo mogli odrediti tačnost rezultujućih ko-trening klasifikatora. U tabeli 11 su predstavljene izračunate vrednosti za meru entorpije predložene u [Salaheldin 2010] (jednačina 62) za čije računanje nisu neophodne validacione instance. Ovde je za entropiju *i*-tog pogleda $E(V_i)$ uzeta prosečna vrednost entropije *i*-tog pogleda nastalog slučajnim podelama obeležja.

Skup podataka	$E(V_1)$	$E(V_2)$	$E(V_1*V_2)$	***maxDiv***
News2x2	3.97	4.07	10.9	8.78
LingSpam	15.3	15.8	39.6	31.9
Hepatitis	4.63	4.20	8.60	9.86
Breast-cancer	22.2	20.7	33.1	44.3
Diabetes	56.3	54.7	88.8	112.2
Cylinder-bands	36.6	35.4	59.2	73.2

Tabela 11 Prosečne entropije pogleda nastalih slučajnim podelama obeležja i njihova različitost izračunata prema jednačini (62). Notacija: $E(V_1)$: prosečna entropija prvog klasifikatora na test skupu; $E(V_2)$: prosečna entropija drugog klasifikatora na test skupu; $E(V_1*V_2)$: prosečna entropija kombinovanog klasifikatora na test skupu; *maxDiv*: kombinacija procenjene snage i različitosti individualnih klasifikatora koja se računa prema jednačini (62).

klasifikatora se u slučaju *MV*, *RSSalg* i *IMCC* postavki sastoji od više ko-trening klasifikatora.

Iz tabele 11 se vidi da je je procena pomoću entropije razumno dobra. Mera se razlikuje od procene pomoću *Div_st* mere kod LingSpam skupa podataka za koji je procenjeno da je manje redudantan od Hepatitis skupa podataka i takođe kod Diabetes skupa podataka, za koji je pomoću ove mere procenjeno da je manje redudantan od Cylinder-bands skupa podataka. Ovaj manji eksperiment daje indikaciju da bi se ovakva mera mogla uspešno koristiti kao mera mogućnosti uspeha *RSSalg* postavke.

Pokušaćemo još da procenimo koliko su ko-trening klasifikatori klasifikatori u formiranoj grupi ko-trening klasifikatora međusobno različiti. Na osnovu formule (66) izračunata je *Div* mera koja se odnosi na raznolikost formirane grupe ko-trening klasifikatora korišćene u *MV*, *RSSalg* i *IMCC* postavkama. Dobijeni rezultati su predstavljeni u tabeli 12.

Skup podataka	RSSalg	RSSalg best	IMCC	$Acc(V_1*V_2*...*V_m)$	$Avg(Acc)$	*Div*
News2x2	89.1±3.1	**90.6±1.8**	85.7±0.7	0.859	0.803	0.056
LingSpam	88.5±7.0	91.1±5.9	**95.3±0.5**	0.802	0.765	0.037
Hepatitis	82.6±3.5	**86.5±3.4**	85.8±5.1	0.805	0.848	0.043
Breast-cancer	67.0±6.8	70.4±5.3	**73.9±2.2**	0.683	0.669	0.013
Diabetes	63.9±3.7	67.7±1.8	**71.7±2.1**	0.651	0.625	0.026
Cylinder-bands	54.3±4.8	61.6±2.5	**65.9±1.9**	0.519	0.527	0.008

Tabela 12 Performanse *RSSalg*, *RSSalg_best i IMCC* postavke na različitim skupovima podataka (deo rezultata kopiran iz tabele 2 iz odeljka 4.4) i različitost u grupi od *m* formiranih ko-trening klasifikatora korišćenih u *MV*, *RSSalg* i *IMCC* postavci. Notacija: $\boldsymbol{Acc(V_1*V_2*...*V_m)}$: tačnost kombinovanog klasifikatora; $Avg(Acc)$: prosek tačnosti pojedinačnih klasifikatora; *Div*: različitost izračunata prema formuli (66).

Iz tabele 12 vidimo da postoji korelacija izmerenih vrednosti različitosti u skupu podataka i performansi *RSSalg* postavke. Potencijalno, korišćenjem navedenih mera bi se u budućnosti mogle povećati performanse *RSSalg* postavke time što bi se odabir korišćenih podela obeležja (koji je trenutno u potpunosti slučajan) mogao zameniti nekim metodološkim pristupom odabira slučajnih podela koje će biti korišćene u cilju povećanja raznolikosti grupe klasifikatora. Naime, algoritam predložen u [Opitz 1999] selektuje podskupove obeležja originalnog skupa i na osnovu svakog dobijenog podskupa trenira odvojeni klasifikator. Koji će tačno podskupovi obeležja biti korišćeni za treniranje klasifikatora se određuje korišćenjem predložene mere za računanje međusobne raznolikosti klasifikatora u skupu podataka (jednačina 65) u okviru genetskog algoritma. Teži se da formirana grupa klasifikatora bude što raznovrsnija. Prilagođavanjem ove mere za ko-trening postavku gde nemamo anotirane instance da odredimo tačnost klasifikatora, zasnovano na principu predloženom u [Salaheldin 2010], mogli bi smo odrediti slučajne podele koje će rezultovati većom različitošću u grupi u odnosu na to da su date podele birane na slučajan način. Moguće je da bi smo ovim pristupom mogli istovremeno optimizovati i broj slučajnih podela obeležja koji vredi koristi prilikom

formiranja modela, odnosno utvrditi tačku nakon koje dalje dodavanje klasifikatora u grupu neće značajno doprineti porastu performansi.

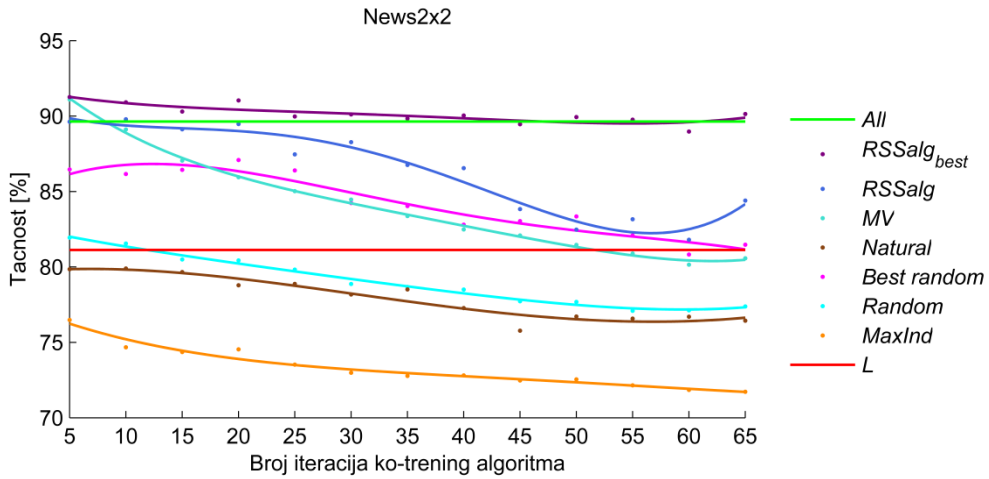
4.5.4 Uticaj broja iteracija ko-treninga

Ranije studije pokazuju da je ko-trening osetljiv na broj iteracija [Pierce 2001; Ng 2003]. U originalnoj verziji ko-treninga [Blum 1998] korišćen je predefinisan broj iteracija i mnogi autori prate ovu praksu. Jedna od korišćenih varijacija ko-treninga jeste da se ko-trening proces nastavi sve dok svi korišćeni neanotirani podaci budu anotirani ko-trening procesom [Nigam 2000b]. U [Chan 2004b] je predložena još jedna varijacija ko-treninga gde se koristi validacioni skup za određivanje tačke u kojoj su performanse maksimizovane. U ovoj postavci se ko-trening pušta da iterira sve dok validacioni skup ne nagovesti da je dostignut maksimum performansi ili dok ne ponestane neanotiranih primera.

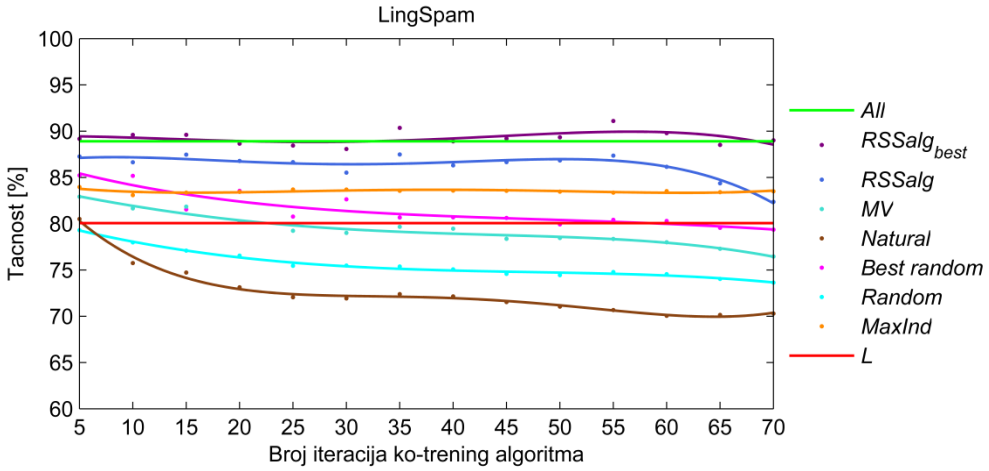
U praksi je primećeno da puštanjem ko-trening algoritma da iterira do konvergencije može doći do degradacije u performansama klasifikatora [Pierce 2001]. U [Wang 2007] je izneto teorijsko objašnjenje ovog fenomena: nakon određenog broja iteracija ko-trening ne može dalje da unapredi performanse zbog toga što dva klasifikatora postaju sve sličnija. Zbog toga, ako se ko-trening proces nastavi do konvergencije, povećava se verovatnoća da klasifikatori prave iste greške i, kako se ovi klasifikatori kombinuju u cilju davanja finalne predickije, ovakve greške će se pojačavati. Zbog toga dolazi do prevelikog prilagođavanja podacima (*overfiting*), što dovodi do smanjenja performansi. Autori takođe predlažu postupak određivanja trenutka kada je potrebno prestati sa iteracijama u ko-trening algoritmu. Međutim, ovaj postupak se zasniva na meri moći generalizacije klasifikatora koja se može tačno izmeriti jedino pomoću anotiranog validacionog skupa, koji u praktičnim primenama nije dostupan. U izvedenim eksperimentima [Wang 2007] ova vrednost je aproksimirana. Pokazano je da ovaj pristup ne daje loše rezultate, ali da usled uvedene aproksimacije ipak nije dostignuta očekivana tačnost, zbog čega je unapređenje ove aproksimacije zadatak za budućnost.

U ovom odeljku analiziran je uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki. Eksperiment je izvršen merenjem performansi rešenja nakon svake pete iteracije ko-treninga, a ko-trening je pušten da iterira sve dok u neanotiranom skupu ne nestane podataka za anotaciju. Svi parametri sem broja iteracija i broja korišćenih slučajnih podela su u ovom eksperimentu fiksirani na vrednosti izlistane u odeljku 4.2. Za broj korišćenih slučajnih podela uzete su minimalne vrednosti nakon koje dalje slučajne podele ne utiču u značajnoj meri za rezultat, izlistane u tabeli 8. Eksperiment nije izvršen na skupu podataka Hepatitis, pošto je ovaj skup ima suviše mali broj instanci za ovo testiranje (uz vrednosti parametara p i n definisane u odeljku 4.2 moguće je izvrštiti oko 5 iteracija).

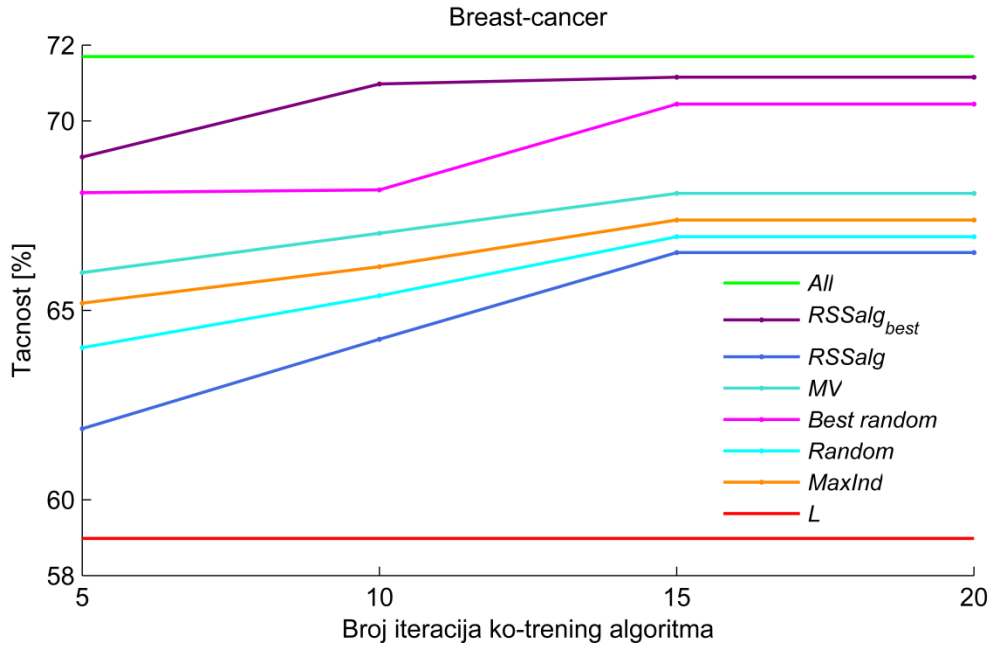
Rezultati ovih eksperimenata za različite skupove podatka su grafički prikazani na grafikonima 13 – 17. Horizontalna osa na ovim graficima predstavlja broj iteracija ko-trening algoritma, a vertikalna osa predstavlja tačnost postignutu od strane različitih algoritama (u procentima). Analizirane su iste postavke kao u odeljku 4.5.1. Performanse postavki *All* i *L* na koje ne utiče broj iteracija ko-treninga su predstavljene ravnim linijana, a performanse *Random*, *Best random*, *MV*, *RSSalg* i $RSSalg_{best}$ i $MaxInd_{best}$ podela su predstavljene tačkama. Kao vodilja, za ove podele je nacrtan i polinom fitovan na dobijene vrednosti (izuzev za Breast-cancer skup podataka gde je broj merenja suviše mali).



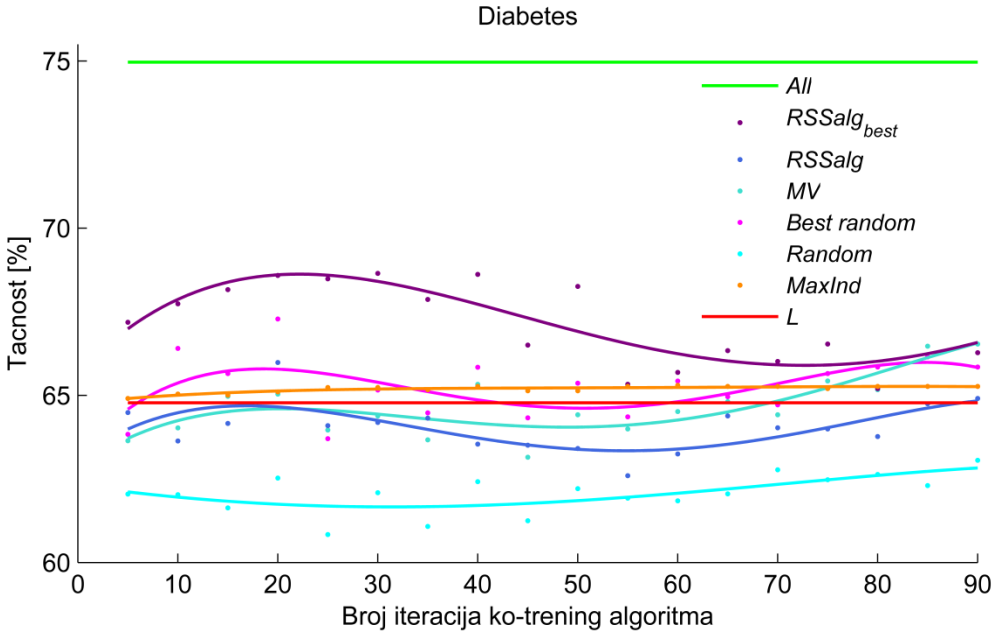
Grafikon 13 Uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki na News2x2 skupu podatka.



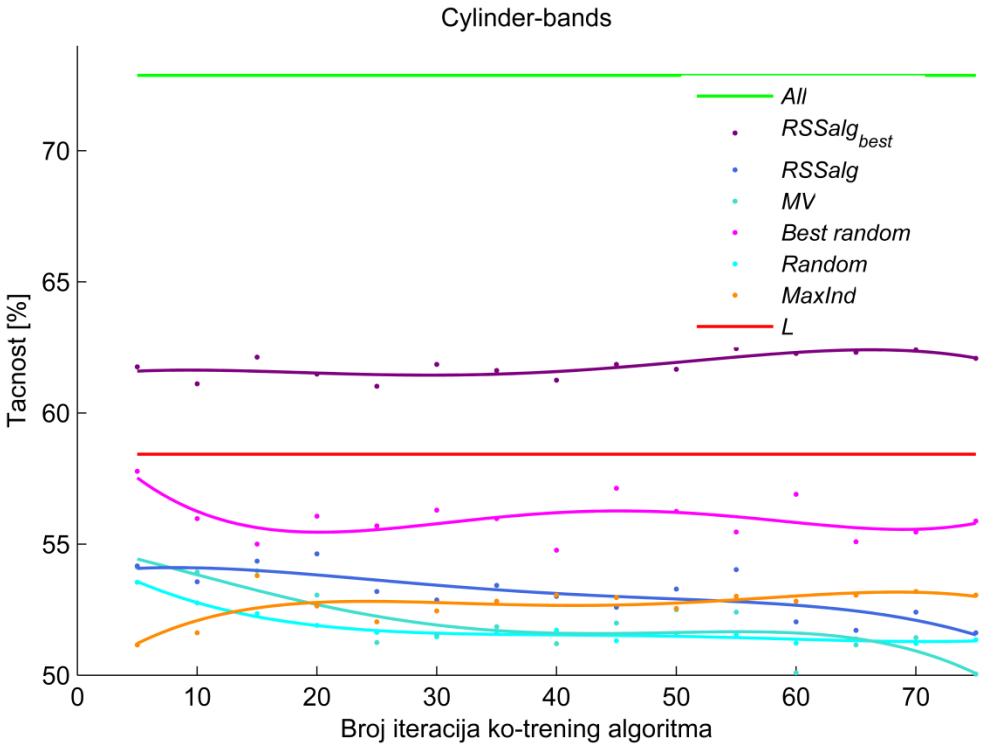
Grafikon 14 Uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki na LingSpam skupu podatka.



Grafikon 15 Uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki na Breast-cancer skupu podatka.



Grafikon 16 Uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki na Diabetes skupu podatka.



Grafikon 17 Uticaj broja iteracija ko-trening algoritma na performanse testiranih postavki na Cylinder-bands skupu podatka.

U tabeli 13 je prikazana standardna devijacija, a u tabeli 14 razlika minimalne i maksimalne izmerene tačnosti postavki na različitim skupovima podataka.

Skup podataka	**Natural**	**Random**	**Best Random**	**MaxInd best**	**MV**	**RSSalg**	**RSSalg best**
News2x2	1.44	1.67	2.13	1.37	3.41	2.90	.634
LingSpam	2.82	1.60	1.94	.201	1.81	1.40	.798
Breast-cancer		1.41	1.33	1.06	1.00	2.22	1.03
Diabetes		.584	.930	.111	.914	.741	1.19
Cylinder-bands		.647	.820	.660	1.30	.937	.464
	2.13	1.18	1.43	0.681	1.69	1.64	.823

Tabela 13 Standardnadevijacija izmerene tačnosti postavki na različitim skupovima podataka.

Skup podataka	Natural	Random	Best Random	MaxInd best	MV	RSSalg	RSSalg best
News2x2	4.13	4.85	6.26	4.76	11.0	7.99	2.29
LingSpam	10.5	5.67	5.86	.880	6.46	5.12	3.03
Breast-cancer		2.93	2.34	2.19	2.09	4.65	2.11
Diabetes		2.22	3.58	0.39	3.38	3.38	3.45
Cylinder-bands		2.33	3.01	2.64	4.07	3.01	1.44
prosek	7.32	3.60	4.21	2.17	5.40	4.83	2.46

Tabela 14 Razlika maksimalne i minimalne izmerene tačnosti postavki na različitim skupovima podataka [%].

Na osnovu prikazanih rezultata možemo zaključiti sledeće:

- Uvećavanjem broja iteracija ko-treninga je zaista uočeno opadanje performansi svih ko-trening postavki za većinu skupova podataka. Izuzetak je Breast-cancer skup podataka, najverovatnije zbog svoje male veličine, te i malog broja instanci koji se dodaje u inicijalni obučavajući skup čak i u slučaju kada se ko-trening pusti da iterira sve dok ne anotira sve instance neanotiranog skupa. Takođe, i Diabetes skup podataka je izuzetak u smislu da performanse postavki ponovo kreću da rastu nakon 60. iteracije.
- Najveća izmerena tačnost je kod većine postavki na većini skupova podataka bila negde oko 20. iteracije.
- Najmanju osetljivost na broj iteracija su pokazale $RSSalg_{best}$ postavka i $MaxInd_{best}$ postavka. Uspeh $MaxInd_{best}$ postavke može ležati u izboru korišćenih klasifikatora, budući da je najstabilnije performanse pokazala na LingSpam skupu podataka gde su kao unutrašnji klasifikatori korišćene RBF mreže, kao i na Cylinder-bands i Diabetes skupovima podataka gde su kao unutrašnji klasifikatori korišćene mašine potpornih vektora. Sledeća po stabilnosti je *Random* postavka, međutim tu je potrebno napomenuti da tačnost *Random* postavke predstavlja uprosečenu tačnost više slučajnih podela, tako da efekat „glatkoće“ može da proizilazi iz toga. Nakon *Random* postavke, po stabilnosti idu *RSSalg*, *MV* i *Natural* postavka, redom.

U navedenim zaključcima navedene su i dve stavke koje zahtevaju dodatnu analizu koja izlazi iz opsega ove disertacije, ali predstavljaja dobar pravac za buduća istraživanja: rast performansi ko-trening algoritma sa povećanjem broja iteracija na Diabetes skupu podataka i zapažanje da stabilnost postavki na parametre ko-treninga može da zavisi i od korišćenih unutrašnjih klasifikatora.

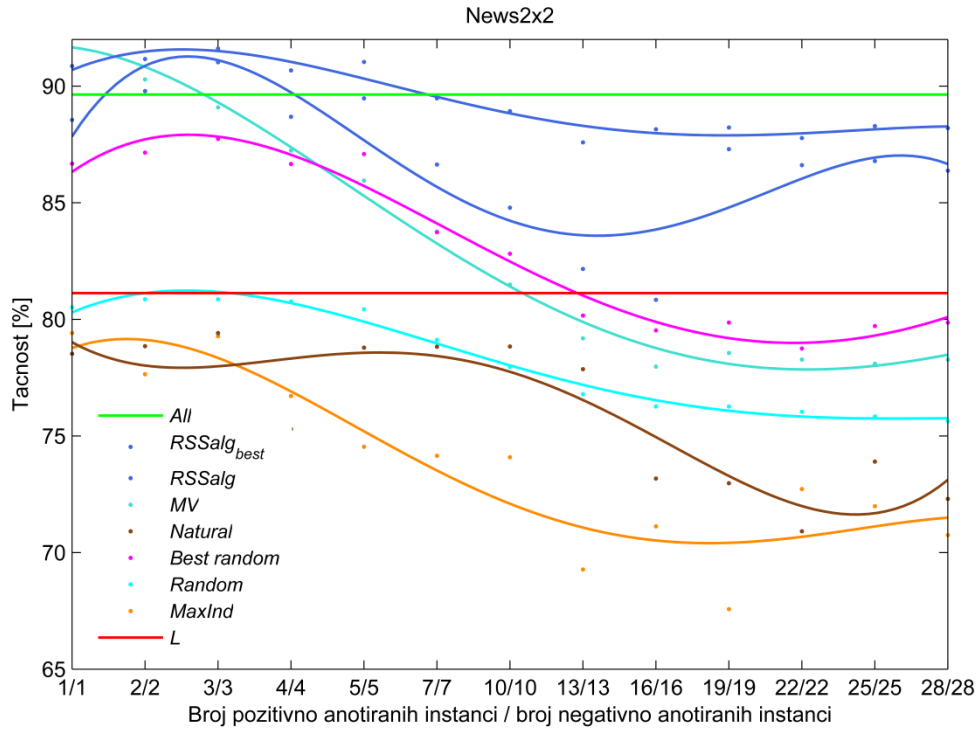
4.5.5 Uticaj veličine rasta anotiranog skupa

Pod veličinom rasta se ovde misli na broj najpouzdanije anotiranih instanci koje se dodaju u inicijalni anotirani skup u svakoj iteraciji ko-trening algoritma. U originalnoj postavci ko-treninga [Blum 1998] korišćen je

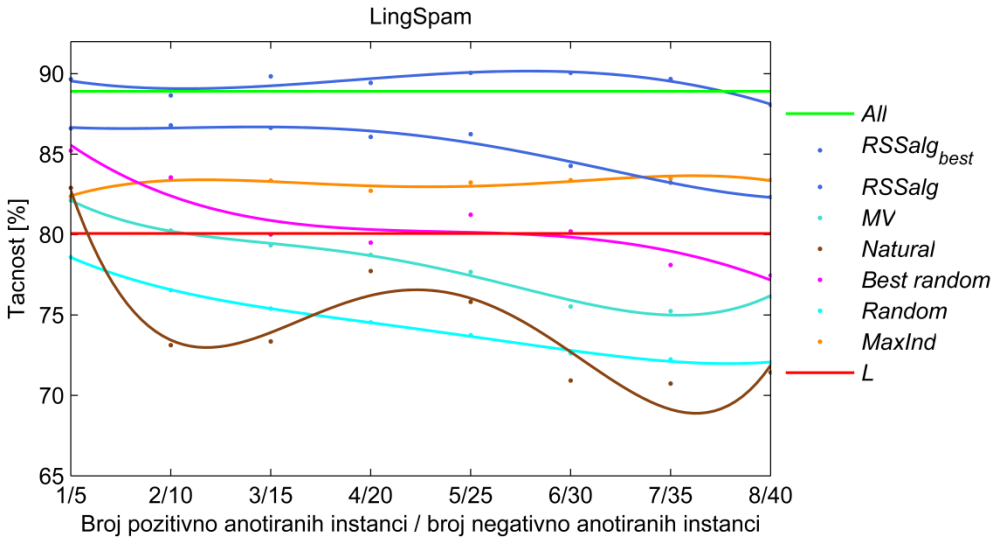
predefinisan broj instanci po svakoj klasi koji će se dodati anotoranom skupu u svakoj iteraciji ko-treninga (parametri *p* i *n*). Ovi parametri bi trebali biti odabrani tako da njihov odnos grubo reflektuje pravu proporciju klasa u podacima. U [Munkhdalai 2012] autori su testirali osetljivost ko-treninga na veličinu rasta anotiranog skupa u svakoj iteraciji. Pokazalo se da dodavanje većeg broja instanci dovodi do brzog povećanja performansi u nekoliko iteracija, ali i da previše velika količina primera degradira performanse. Ukoliko se u svakoj iteraciji instance dodaju restriktivnije, pobojšanje performansi je manje, ali takođe ne dolazi do opadanja performansi nakon određene iteracije.

U ovom odeljku analiziran je uticaj parametra veličine rasta ko-trening algoritma na performanse testiranih postavki. U eksperimentima su izvršena merenja performansi poređenih postavki za različite *p*/*n* vrednosti tako što je odabiran parametar *p*, nakon čega je parametar *n* računat tako da odnos *p*/*n* grubo reflektuje distribuciju klasa u podacima, kao što je preporučeno u [Blum 1998]. Parametri *p* i *n* su povećavani sve dok se nije došlo do toga da gotovo sve instance podskupa *u'* budu anotirane u jednoj iteraciji. Svi ostali parametri, izuzev broja korišćenih slučajnih podela, su u ovom eksperimentu fiksirani na vrednosti izlistane u odeljku 4.2. Za broj korišćenih slučajnih podela uzeta je minimalna vrednost nakon koje dalje slučajne podele ne utiču u značajnoj meri za rezultat (tabela 8).

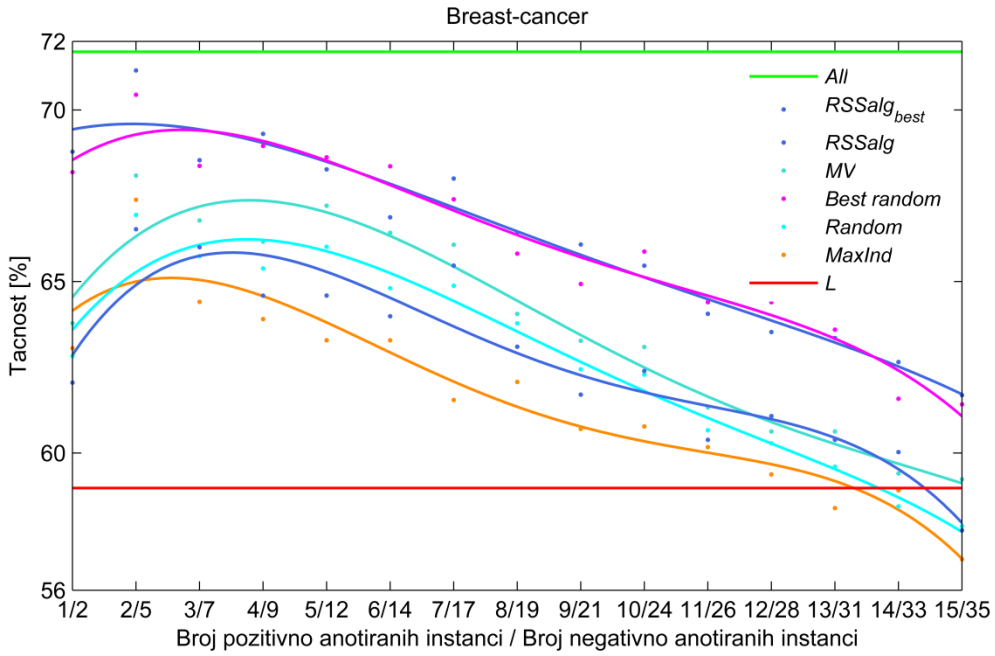
Rezultati izvršenih eksperimenata za različite skupove podatka su grafički prikazani na grafikonima 18 - 23. Horizontalna osa na ovim graficima predstavlja odnos broja pozitivno anotiranih primera (*p*) i broja negativno anotiranih primera (*n*) u svakoj iteraciji ko-treninga u formatu *p*/*n*. Analizirane postavke predstavljene odeljku 4.5.1. Performanse postavki *All* i *L* na koje ne utiče parametar veličine rasta ko-treninga su predstavljene ravnim linijana, a performanse *Random*, *Best random*, *MV*, *RSSalg* i $RSSalg_{best}$ i $MaxInd_{best}$ podela su predstavljene tačakama. Kao vodilja, za ove podele je nacrtan i polinom fitovan na dobijene vrednosti.



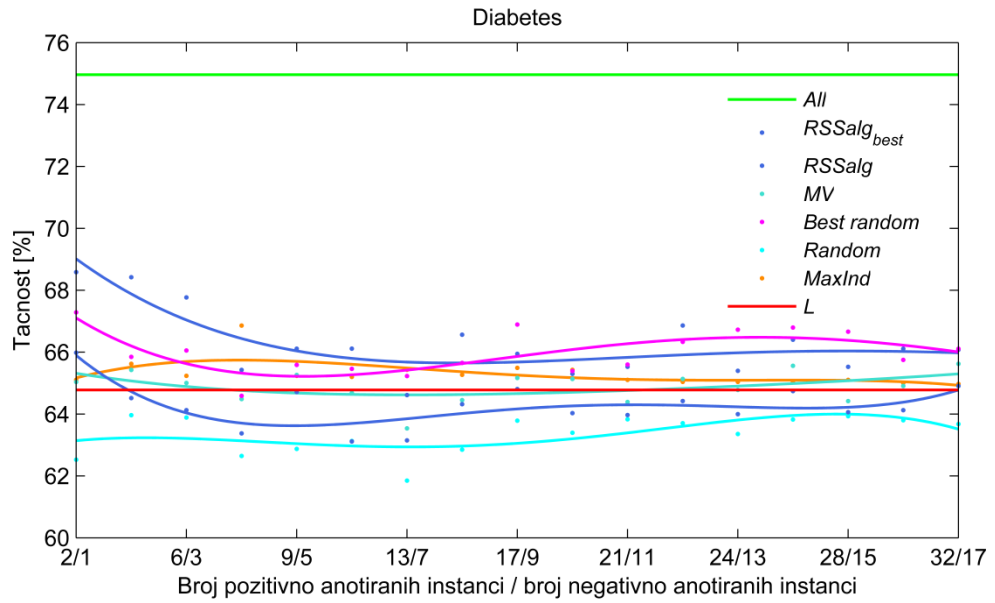
Grafikon 18 Uticaj parametra veličine rasta na performanse testiranih postavki na News2x2 skupu podatka.



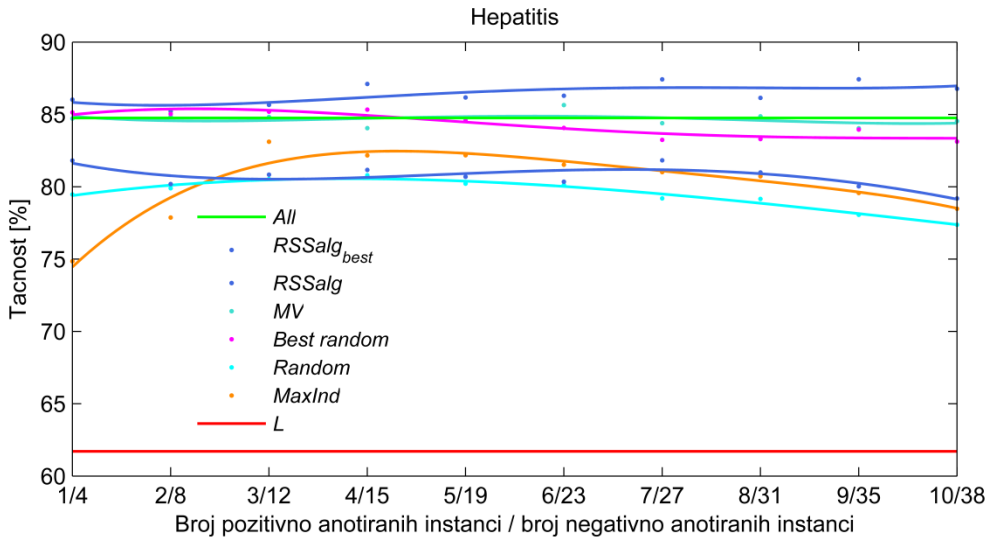
Grafikon 19 Uticaj parametra veličine rasta na performanse testiranih postavki na LingSpam skupu podatka.



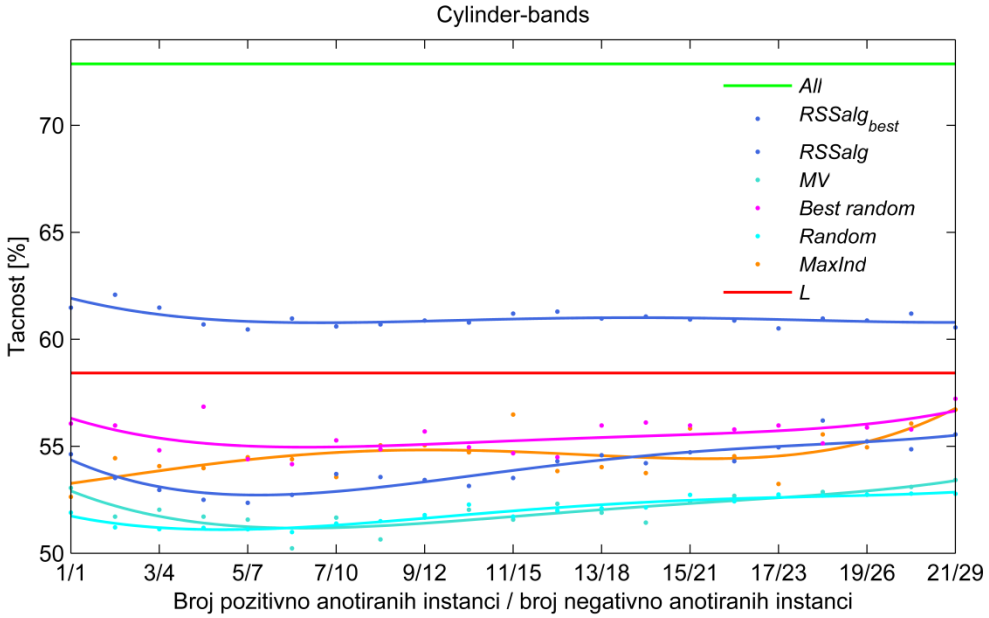
Grafikon 20 Uticaj parametra veličine rasta na performanse testiranih postavki na Breast-cancer skupu podatka.



Grafikon 21 Uticaj parametra veličine rasta na performanse testiranih postavki na Diabetes skupu podatka.



Grafikon 22 Uticaj parametra veličine rasta na performanse testiranih postavki na Hepatitis skupu podatka.



Grafikon 23 Uticaj parametra veličine rasta na performanse testiranih postavki na Cylinder-bands skupu podatka.

U tabeli 15 je prikazana standardna devijacija, a u tabeli 16 razlika minimalne i maksimalne izmerene tačnosti postavki na različitim skupovima podataka.

Skup podataka	Natural	Random	Best Random	MaxInd best	MV	RSSalg	RSSalg best
News2x2	3.09	2.20	3.57	3.72	5.24	2.92	1.48
LingSpam	4.18	2.30	2.62	.400	2.44	1.75	.710
Breast-cancer		2.28	2.74	2.71	2.97	2.52	2.77
Diabetes		.614	.705	.443	.522	.699	1.09
Hepatitis		1.08	.873	2.50	.462	.813	.744
Cylinder-bands		.641	.799	1.05	.802	1.03	.386
Prosek	3.63	1.52	1.88	1.80	2.07	1.62	1.20

Tabela 15 Standardnadevijacija izmerene tačnosti postavki na različitim skupovima podataka.

Skup podataka	Natural	Random	Best Random	MaxInd best	MV	RSSalg	RSSalg best
News2x2	8.5	5.22	8.99	11.8	14.0	10.2	4.02
LingSpam	12.2	6.55	7.76	1.11	6.88	4.46	2.00
Breast-cancer		9.07	9.03	10.5	8.85	8.78	9.47
Diabetes		2.11	2.69	1.89	2.09	2.87	3.97
Hepatitis		3.42	2.21	8.26	1.60	2.64	2.22
Cylinder-bands		1.80	3.06	4.07	3.19	3.84	1.62
Prosek	10.35	4.69	5.62	6.27	6.10	5.46	3.88

Tabela 16 Razlika maksimalne i minimalne izmerene tačnosti postavki na različitim skupovima podataka [%].

Na osnovu prikazanih rezultata možemo zaključiti sledeće:

- Za većinu skupova podataka je uočeno da nakon određene veličine rasta performanse ko-treninga opadaju. Izuzetak je Cylinder-bands skup podataka gde su performanse svih postavki relativno robustne u odnosu na veličinu rasta.
- Nije uočena generalna proeporuka za odabir ovog parametra, osim da veličina skupa primera koja se dodaje u svakoj iteraciji ne treba da bude prevelika.
- Najmanju osetljivost na izbor ovog parametra je pokazala $RSSalg_{best}$ postavka, nakon čega slede redom *Random*, *RSSalg*, $MaxInd_{best}$, *Best random*, *MV* i *Natural* postavke. Za *Random* ponovo važi da uočena „glatkoća“ može biti posledica uprosečavanja tačnosti za više različitih podela.

Detaljna analiza navedenih zaključaka zahtevaja dodatne eksperimente i izlazi iz opsega ove disertacije, ali takođe predstavljaja dobar pravac za buduća istraživanja.

5 Softverska arhitektura

U ovom poglavlju predstavljena je implementacija prototipova modela predstavljenih u disertaciji. Implementacija dela sistema, vezana za izvršavanje eksperimenata sa osnovnim ko-trening algoritmom i *MaxInd* postavkom, je dobijena od autora rada [Feger 2008]. Autor disertacije je na ovaj postojeći sistem dodao neophodnu funkcionalnost za izvršavanje modela predstavljenih u ovoj disertaciji.

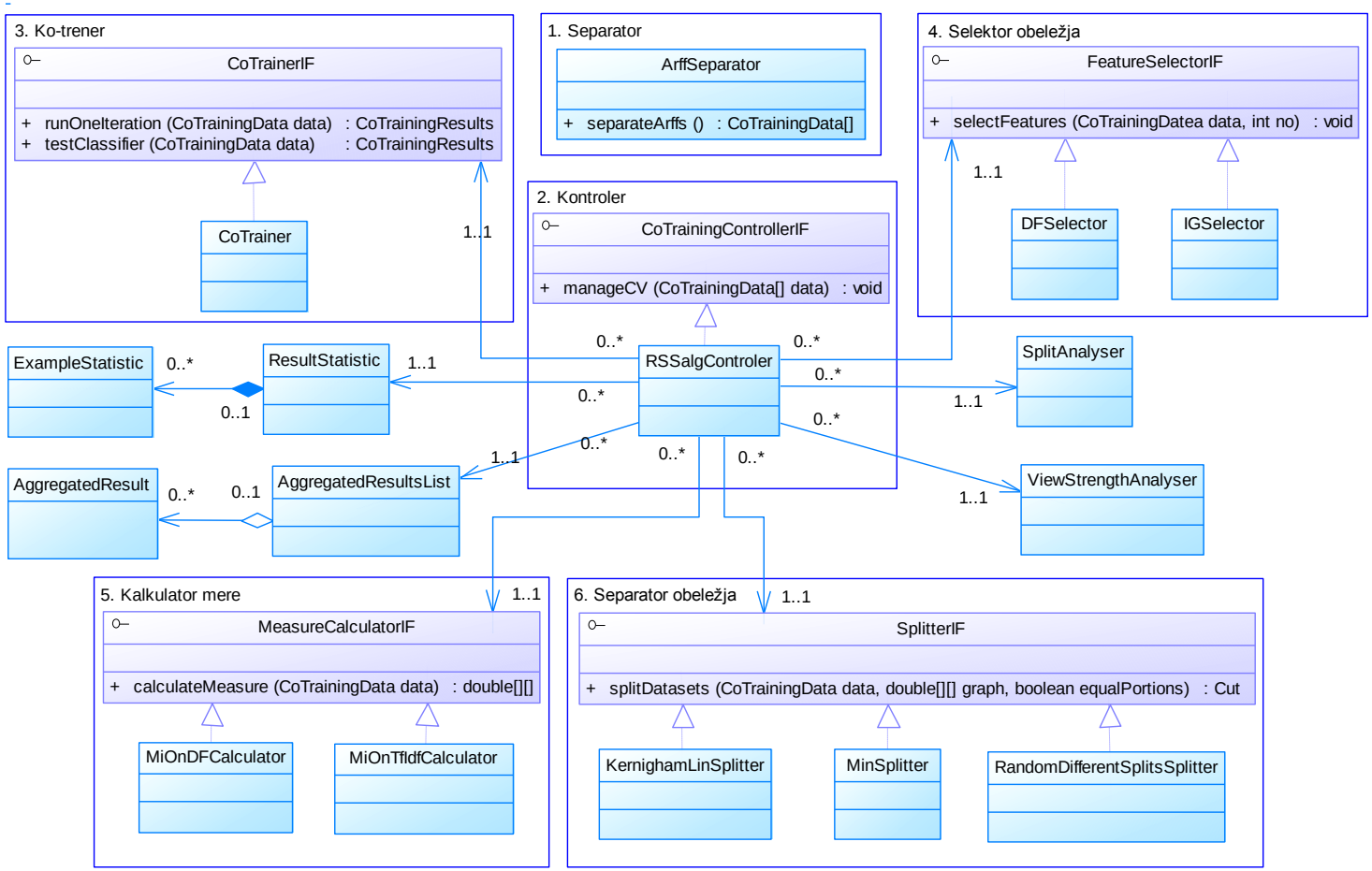
Sistem je dizajniran na modularan način kako bi se mogao što lakše konfigurisati za različite eksperimente. Na slici 13 prikazan je UML dijagram najvažnijih klasa sistema. Sistem se sastoji od šest glavnih delova: separator, kontroler, ko-trener, selektor obeležja, kalkulator mere i separator obeležja. Sem ovih modula, izdvojene su i klase za analizu i pisanje dobijenih rezultata.

Prilikom pokretanja programa, učitavaju se parametri eksperimenta definisani u dve .properties datoteke:

1. Parametri vezani za korišćeni skup podatka (lokacija skupa podataka koji je definisan u *ARFF* formatu, broj delova *n* na koji će se skup podeliti radi izvršenja unakrsne validacije, imena kategorija, naziv klasnog obeležja u skupu podataka, itd.);
2. Paramtri vezani za detalje ko-trening postavke koja se izvršava (lokacija na kojoj će se čuvati rezultati eksperimenta, parametri ko-trening algoritma, broj korišćenih slučajnih podela, parametri genetskog algoritma za optimizaciju pragova ukoliko je u pitanju *RSSalg* postavka, itd).

Na početku izvršavanja programa pokreće se modul separator (predstavljen klasom *ArffSeparator*). Najvažnija metoda ovog interfejsa je *separateArffs* metoda koja učitava skup podataka i deli ga na *n* približno jednakih delova koji će se koristiti u unakrsnoj validaciji. Dobijeni delovi se koriste da se za svaki od *n* krugova unakrsne validacije definiše inicijalni anotirani skup, skup neanotiranih podataka i test skup. Podaci koji odgovaraju jednom krugu unakrsne validacije su agregirani u *CoTrainingData* objektu. Ovaj objekat pored podele na anotirani, neanotirani i test skup sadrži i definiciju podele obeležja koja će se koristiti za ko-trening algoritam. Rezultat *separateArffs* metode je niz *CoTrainingData* objekata koji se prosleđuju kontroleru.

Kontroler (predstavljen interfejsom *CoTrainingControllerIF*) je centralni deo programa koji je zadužen za koordinaciju ostalim modulima. Najvažnija metoda ovog interfejsa je *manageCV* koja prima niz *CoTrainingData* objekata i vrši eksperiment unakrsne validacije. U okviru ove metode se na osnovu zadatih parametara eksperimenta učitavaju i izvršavaju potrebni moduli. Nakon završetka eksperimenta, rezultati se agregiraju i zapisuju u datoteke.



Slika 13 Dijagram klasa sistema predstavljenog u disertaciji.

Modul koji se uvek izvršava jeste ko-trener (predstavljen interfejsom *CoTrainerIF*). Ovaj modul izvršava ko-trening algoritam (algoritam 1 u odeljku 1.3.4) iteraciju po iteraciju i na kraju svake iteracije prijavljuje kontroleru sve promene nastale na skupu podataka, kao i trenutne performanse klasifikatora. Dve najvažnije metode ovog interfejsa su *runOneIteration* i *testClassifier*. Metoda *runOneIteration* prima objekat tipa *CoTrainingData* i ažurira ga prema izvršenoj iteraciji ko-treninga. Metoda *testClassifier* prima *CoTrainingData* objekat, obučava klasifikator na anotiranim podacima datog *CoTrainingData* objekta i primenjuje ga na test podatke istog *CoTrainingData* objekta. Obe metode kao rezultat vraćaju *CoTrainingResults* objekat koji reprezentuje rezultat izvršavanja ko-trening algoritma. Ovaj objekat sadrži matricu kontingencije dobijenu primenom finalnih ko-trening klasifikatora na test skup i obezbeđuje funkcionalnost računanja različitih mera (tačnost, *f*-mera, itd). Takođe, ovaj objekat sadrži i rezultate dobijene u pojedinačnim iteracijama.

Preostali moduli su opcioni i učitavaju se samo ukoliko su neophodni za izvršenje određenog eksperimenta. Funkcionalnost ovih modula je propisana zadatim interfejsom, a za svaki postoji nekoliko različitih implementacija. Konkretna implementacija koja će biti korišćena u eksperimentu se propisuje kroz parametre vezane za detalje ko-trening postavke koja se se izvršava. Implementacija se definiše punim, kvalifikovanim imenom korišćene klase, a sam kontroler na osnovu imena pravi objekat zadate klase. Na ovaj način nova implementacija tražene klase može biti korišćena bez potrebe za promenom implementacije kontrolera. Opcioni moduli su:

1. Selektor obeležja (interfejs *FeatureSelectorIF*). Ovaj modul je zadužen za redukciju dimenzionalnosti skupa podataka. Na primer, za svaki od skupova podataka prirodnog jezika je u eksperimentima zadržano svega 200 obeležja koja su po zadatoj meri procenjena kao najvažnija. Osnovna metoda ovog interfejsa je *selectFeatures* koji prima *CoTrainingData* objekat i broj obeležja koje je potrebno zadržati. Metoda *selectFeatures* ažurira skupove podataka u *CoTrainingData* objektu u skladu sa izvršenom selekcijom obeležja. U sistemu postoje dve klase koje implementiraju interfejs *FeatureSelectorIF*: *DFSelector* i *IGSelector*, koje vrše selekciju obeležja koristeći frekvenciju pojavljivanja termina (*term*) u dokumentu (*DF*, *document frequency*) i informacioni dobitak (*IG*, *information gain*) [Yang 1997] kao meru, respektivno.
2. Kalkulator mere (interfejs *MeasureCalculatorIF*). Ovaj modul je zadužen za izračunavanje mere koja će se koristiti za optimizaciju podele obeležja na poglede. Najvažnija metoda ovog interfejsa je *calculateMeasure*. Ova metoda prima objekat tipa *CoTrainingData* i na osnovu prosleđenih podataka izračunava težinski graf gde su obeležja predstavljena čvorovima grafa, a težina grane koja povezuje dva čvora predstavlja meru međusobne zavisnosti odgovarajućih obeležja. U sistemu ovaj interfejs implementiraju klase *MIOnDFCalculator* i *MIOnTfIdfCalculator*. Ove klase međusobnu zavisnost

obeležja računaju koristeći meru uzajamna informativnost (*Mutual Information, MI*) [MacKay 2003]. Međusobno se razlikuju po tome da li se ova mera primenjuje na katergorička obeležja (*MIOnDFCalculator*) ili se primenjuje na kontinualna obeležja (*MIOnTfIdfCalculator*). U slučaju kontinualnih obeležja, vrednosti atributa se pre računanja *MI* mere diskretizuju.

3. Separator obeležja (interfejs *SplitterIF*). Najvažnija metoda ovog interfejsa je *splitDataset*. Ova metoda prima graf koji je izračunat od strane kalkulatora mere, podatke i indikator da li dva pogleda treba da budu približno iste veličine. Metoda na osnovu prosleđenog grafa vrši podelu obeležja i u skladu sa njom ažurira *CoTrainingData* objekat. Povratna vrednost metode je *Cut* objekat koji reprezentuje datu podelu obeležja (sadrži sumu presečenih grana grafa i listu obeležja jednog od dva pogleda), a koja se može koristiti za različite analize date podele. U sistemu *SplitterIF* implementiraju tri klase: *KernighamLinSplitter*, *MinSplitter* i *RandomDifferentSplitsSplitter*. Ove klase se međusobno razlikuju po načinu podele grafa obeležja na dva podgrafa. Klase *KernighamLinSplitter* i *MinSplitter* su dizajnirane da podele graf na dva podgrafa tako da je suma težina presečenih grana grafa minimalna. Klasa *KernighamLinSplitter* koristi heuristički pristup predložen u [Kernigham 1970] kojim je omogućena podela grafa na dva približno jednaka dela. Klasa *MinSplitter* za podelu grafa koristi jednostavan algoritam predložen u [Karger 1996] koji ne vodi računa o tome da dobijeni podgrafovi budu približno jednaki (parametar *equalPortions* nema uticaja na ovu implementaciju). Usled toga, pogledi koji rezultuju primenom algoritma implementiranog u *MinSplitter* klasi mogu biti veoma nebalansirani (jedan pogled može sadržati značajno više obeležja od drugog pogleda i zbog toga imati značajno bolje performanse). Klasa *RandomDifferentSplitsSplitter* deli čvorove grafa u dva podgrafa na slučajan način. Težine grana grafa u ovoj implementaciji nemaju nikakav uticaj. Klasa takođe vodi računa o tome da se ne ponove dve iste slučajne podele grafa.

Preostale važne klase sistema su:

- *ViewStrengthAnalyser* – klasa koja izračunava jačinu pogleda, kako na *L* skupu (anotirani podaci), tako i na *All* skupu (anotirani podaci i podaci iz neanotiranog skupa sa pridodatim tačnim anotacijama). Jačinu pogleda je moguće izraziti putem više mera (tačnost, *f*-mera, itd)
- *SplitAnalyser* – izračunava mere za evaluaciju date podele obeležja: sumu težina presečenih grana grafa i sumu težina grana grafa u svakom od pogleda ponaosob.
- *AggregatedResultsList* – objekat koji agregira rezultate različitih izvršenih eksperimenata. Predstavlja listu objekata *AggregatedResults*. Objekat *AggregatedResults* predstavlja skup rezultata različitih testiranih ko-trening postavki (*MaxInd*, *Random*, *RSSalg*, itd) koje su testirane sa istim parametrima (ko-trening parametri, kao i broj slučajnih podela). Ovi

objekti omogućavaju lako pisanje/čitanje rezultata u/iz datoteka predstavljenih u XML formatu.

- *ResultStatistic* – Statistika koja se računa u toku izvršavanja *RSSalg* algoritma (algoritam 5, odeljak 3.2). Sastoji se iz liste objekata tipa *ExampleStatistic* koja predstavlja statistiku o jednoj instanci (koja je tokom ko-trening procesa dodata u inicijalni obučavajući skup): koliko puta je datoj instanci dodeljeno koje klasno obeležje. Ove klase omogućavaju lako pisanje/čitanje dobijene statistike u/iz datoteka u XML formatu.

U cilju izvršavanja *RSSalg*, *MV* i *IMCC* postavke, klasa *RSSalgControler* će u svakom krugu unakrsne validacije (za jedan *CoTrainingData* objekat) više puta pozvati module separator obeležja i ko-trener. Za implementaciju separatora obeležja se u ovim postavkama koristi klasa *RandomDifferentSplitsSplitter*. Nakon svakog pokretanja separatora obeležja poziva se ko-trening modul radi izvršavanja ko-treninga. Svako izvršavanje ko-treninga je praćeno ažuriranjem statistike slučajnih podela (implementiranoj u klasi *ResultStatistics*). Takođe, sistem nakon izvršavanja svakog ko-treninga primenjuje dobijeni klasifikator na test skup i beleži predikcije klasifikatora za svaku instancu test skupa. Na ovaj način se dobija datoteka u ARFF formatu gde su za svaku test instancu zabeleđeni njeni atributi, stvarno klasno obeležje, kao i dodatni atributi: svaki od dodatnih atributa predstavlja predikciju jednog ko-trening klasifikatora za datu instancu.

Nakon izvršenja željenog broj podela, u *RSSalg* postavci se pokreće genetski algoritam za optimizaciju pragova. U *MV* postavci se, umesto genetskog algoritma, na sačuvane predikcije klasifikatora primenjuje većinsko glasanje u cilju dobijanja finalne predikcije. Za primenu *GMM-MAPML* algoritma u okviru *IMCC* postavke korišćena je implementacija koja kao ulaz prima opisanu datoteku u ARFF formatu u kojoj su zabeležene predikcije svih ko-trening klasifikatora. U trenutku pisanja disertacije implementacija *GMM-MAPML* postavke nije više bila javno dostupna, a na osnovu specifikacije u radu nije je bilo moguce reimplementirati.

Implementacija sistema predstavljenog u disertaciji se zasniva na Java platformi, uz korišćenje *WEKA* biblioteke za mašinsko učenje[24] [Hall 2009] koja pruža gotove implementacije različitih tehnika mašinskog učenja kao i alate za osnovne zadatke tekstualne kategorizacije. Za osnovne klasifikatore koji se koriste u okviru ko-trening algoritma korišćene su implementacije iz WEKA biblioteke, ali se takođe može primeniti bilo koja implementacija koja implementira *Classifier* interfejs *WEKA* okruženja.

[24] *WEKA* biblioteka za mašinsko učenje se može naći na adresi http://www.cs.waikato.ac.nz/ml/weka/

Između ostalih parametara, u sistemu je moguće zadati i seme (eng. *Seed*) za generator slučajnosti. Ovo omogućava da se se svi eksperimenti mogu u potpunosti reprodukovati.

6 Analiza mogućnosti primene rešenja

U ovom poglavlju biće analizirana mogućnost primene modela predloženih u disertaciji. Odeljak 6.1 prezentuje problem detekcije subjektivnosti i evaluaciju mogućnosti primene predloženih rešenja na ovaj zadatak. Odeljak 6.2 demonstrira kako bi se predložena rešenja mogla primeniti i na višekategorijske klasifikacione probleme. Odeljak 6.3 prikazuje kako bi se modeli predloženi u ovoj disertaciji mogli upotrebiti za rešavanje problema pojave novog korisnika u sistemu za davanje preporuka.

6.1 Detekcija subjektivnosti

Nadgledanje mišljenja korisnika ima primenu u ogromnom broju domena [Pang 2008]. Korisnici imaju tendenciju da odluku o kupovini proizvoda baziraju na preporukama i savetima drugih korisnika. Zbog toga je u interesu proizvođača da prate javna mišljenja u cilju pobojšanja satisfakcije korisnika. Danas, zahvaljujući inernetu, veoma je lako doći do ogromnog broja javnih mišljenja putem blogova, sajtova namenjenih reviziji proizvoda, on-lajn rasprava, socijalnih mreža, itd. Ogromna količina tekstova koji sadrže mišljenje korisnika motiviše problem njihovog automatskog dobavljanja, razumevanja i sumarizacije. Ovaj problem je adresiran od strane klasifikacije sentimenata – posebne oblasti kategorizacije teksta koja se bavi zadatkom klasifikacije dokumenata prema sentimentu ili mišljenju o datoj temi [Pang 2002].

Tekstovi u kojima korisnici prezentuju svoju reviziju proizvoda istovremeno sadrže i činjenice i subjektivni sadržaj. Na primer, revizije filmova se često sastoje od opisa radnje filma (objektivan sadržaj) koje su isprepletane sa subjektivnim izjavama [Pang 2004]. Za određivanje sentimenta subjektivnog dokumenta (npr. da li je mišljenje o temi pozitivno ili negativno) identifikacija subjektivnih delova teksta je od velikog značaja. Prilikom pokušaja odlučivanja o polaritetu ne-subjektivnog sadržaja, sistemi za automatsku detekciju subjektivnosti prave pogrešne odluke. Zbog toga eliminacija ne-subjektivnih delova teksta koja prethodi određivanju polariteta u velikoj meri podiže performanse sistema.

Zadatak detekcije subjektivnosti predstavlja automatsku detekciju delova teksta koji izražavaju sentiment ili mišljenje. Ovaj zadatak se često posmatra kao problem klasifikacije [He 2010]. Mnogi autori predlažu da se prilikom odluke o polaritetu dokumenta koriste informacije o subjektivnosti na nivou rečenice. Tradicionalno, modeli za automatsku identifikaciju subjektivnih rečenica se obučavaju na skupovima podataka koji se sastoje od velikog broja anotiranih rečenica [McDonald 2007][Mao 2006]. Nedostatak ovog pristupa jeste što se do odgovarajučeg obučavajućeg skupa teško dolazi: obučavajući skupovi se formiraju ručnom anotacijom koja je vremenski zahtevna, a takođe i podložna greškama budući da ne postoji uvek jasna granica koja deli

subjektivnost od objektivnosti. Takođe, subjektivnost je veoma zavisna od domena [Pang 2008], pa klasifikatori trenirani na jednom domenu obično omanu prilikom generalizacije na nove domene.

U ovom odeljku biće reči o tome kako se primenom ko-trening baziranih tehnika može olakšati problem mukotrpne ručne anotacije velikog skupa rečenica neophodnog za treniranje kvalitetnog klasifikatora za automatsku detekciju subjektivnosti na nivou rečenice. Automatska detekcija subjektivnosti na nivou rečenica je idealna postavka za primenu ko-treninga (i, uopšte uzev, tehnika polu-nadgledanog obučavanja) jer se danas, zahvaljujući internetu, može lako doći do ogoromne količine neanotiranih rečenica. Rezultati primene sistema predstavljenih u ovoj disertaciji na problem detekcije subjektivnosti u tekstu publikovani su u radu [Slivka 2012a].

Odeljak je organizovan na sledeći način: u odeljku 6.1.1 izložen je pregled literature u kojoj se ko-trening predlaže kao rešenje problema vezanih za obuku sistema za detekciju subjektivnosti. U odeljku 6.1.2 opisana je metodologija primene sistema opisanih u ovoj disertaciji na zadatak detekcije subjektivnosti. Eksperimentalno poređenje modela predstavljenih u ovoj disertaciji sa mogućim alternativama je izvršeno u odeljku 6.1.3. Ovaj odeljak takođe sarži i diskusiju dobijenih rezultata. Konačno, odeljak 6.1.4 zaključuje ovaj odeljak.

6.1.1 Pregled vladajućih stavova i shvatanja u literaturi

Informacije izvučene iz vrsta reči (eng. *Part-of-speech* (*POS*) tagovi) su veoma korisne u analizi sentimenata i istraživanju mišljenja [Pang 2008]. Na primer, prisustvo prideva u rečenici je veoma dobra indikacija subjektivnosti rečenice [Wiebe 2001].

Međutim, na osnovu jednog jedinstvenog prideva je teško odrediti semantičku orijentaciju zbog ograničenosti konteksta. Na primer, pridev „nepredvidiv“ ima negativan sentiment u kontekstu revizije automobila, dok u kontekstu revizije filma „nepredvidiva radnja“ označava pozitivan sentiment [Turney 2002]. Sa ciljem određivanja semantičke orijentacije teksta u [Turney 2002] se prvo vrši automatsko određivanje vrsta reči (*POS tagging*), nakon čega se iz teksta izvlače fraze sastavljene od dve uzastopne reči čije vrste odgovaraju jednom od nekoliko ručno definisanih kombinacija. U [Murray 2010] se sa ciljem detekcije subjektivnosti automatski uče značajni paterni koji su u korelaciji sa subjektivnim iskazima. Ovi paretni su 3-grami reči, odnosno sintaktičke fraze sačinjene od 3 uzastopne jedinice iz teksta gde svaka jedinica predstavlja ili reč ili vrstu reči. Primeri ovakvih trigrama su „zaista dobra ideja“, „zaista dobra NN[25]“, itd. Navedeni radovi su motivisali upotrebu obeležja

[25] NN – imenica.

konstrusianih pomoću vrsta reči i njihovih n-grama kao jednog pogleda na skup podataka u ovde prikazanom sistemu baziranom na ko-treningu.

U literaturi postoji nekoliko primera primene ko-treninga na probleme semantičke klasifikacije. U [Wiebe 2005] ko-trening je primenjen za obučavanje klasifikatora koji može razlikovati subjektivne i objektivne rečenice urdu jezika. Problem kod urdu jezika je što ne postoji dovoljna količina anotiranih resursa neophodnih za razvoj čak i osnovnih alata za procesiranje prirodnog jezika. Model se u [Wiebe 2005] trenira korišćenjem skupa obeležja koji se sastoji od POS tagova, unigrama i reči koje nose emocije. Predloženi ko-trening kasifikator nije baziran poput klasičnog ko-treninga na istom obučavajućem skupu koji se odlikuje podelom na dva pogleda, već na dva različita obučavajuča skupa opisana istim skupom obeležja. Autori zaključuju da uprkos nedostatku resursa za parsiranje urdu jezika, ko-trening i statistički bazirane tehnike poput *tf-idf* mere omogućavaju razvoj vrhunskih klasifikatora subjektivnosti. Međutim, u datom eksperimentu ko-trening je primenjen na skup podataka koji se sastojao od 470 subjektivnih i 4000 objektivnih rečenica, što je i dalje prilično zahtevan skup podataka za ručnu anotaciju.

Još jedan pristup upotrebe ko-treninga za klasifikaciju sentimenata je izložen u radu [Wan 2009]. Cilj ovog rada bila je obuka modela za detekciju sentimenata u kineskom jeziku. Kao inicijalni anotirani skup korišćeni su anotirani tekstovi pisani na engleskom jeziku, mašinski prevedeni na kineski jezik. Skup neanotiranih primera je formiran mašinskim prevodom neanotiranih tekstova sa kineskog na engleski jezik. Ko-trening algoritam je u osnovi koristio dva SVM klasifikatora, gde je jedan od klasifikatora obučen na kineskim tekstovima, a drugi na engleskim tekstovima. Ovi klasifikatori su potom iterativno primenjivani u cilju izgradnje klasifikatora za detekciju subjektivnosti. U svakoj iteraciji, instance iz neanotiranog skupa su anotirane i dodavane u obučavajuči skup ukoliko su se modeli izgrađeni na različitim jezicima slagali oko anotacije. Nakon dodavanja novih primera klasifikatori su ponovo obučavani na uvećanom obučavajučem skupu. Ovaj metod zavisi od velikog korpusa na engleskom jeziku koji se sastoji od 8000 revizija proizvoda sa Amazona (4000 pozitivno i 4000 negativno anotiranih revizija). Kasnije je u [Joachims 2001] primenjen pristup sličan ovom za simultano unapređenje klasifikacije sentimenata kod oba jezika.

6.1.2 Metodologija

U ovom odeljku je opisana metodologija primene modela predstavljenih u ovoj disertaciji, kao i njihovih ko-trening alternativa na problem automatske detekciju subjektivnosti. Predloženim pristupom se detektuje subjektivnost na nivou rečenice.

U cilju pronalaženje najboljeg modela, poređeni su modeli opisani u odeljku 4.3: klasični ko-trening sa „prirodnom“ podelom obeležja (*Natural*

postavka), ko-trening sa slučajnom podelom obeležja (*Random* postavka), većinsko glasanje ko-trening klasifikatora (*MV* postavka), ko-trening sa *maxInd* podelom obeležja ($MaxInd_{best}$ *postavka*) i *RSSalg* algoritam (postavke *RSSalg* i $RSSalg_{best}$).

Radi primene klasičnog ko-trening algoritma na problem detekcije subjektivnosti, neophodno je definisati „prirodnu“ podelu skupa obeležja na poglede koja nije unapred definisana za ovaj skup podataka.

Ranije studije su pokazale da prisustvo određenih vrsta reči u rečenici predstavlja dobru indikaciju da li je rečenica subjektivna ili objektivna [Wiebe 2001]. Zbog toga se kao prvi pogled na jednu instancu skupa podataka (u ovom slučaju rečenicu) koristi informacija o vrstama reči prisutnih u rečenici [Slivka 2012a]. Automatska anotacija postojećih reči odgovarajućim vrstama reči (tzv. POS (*part-of-speech*) tagovima) izvršena je pomoću Stanfordovog parsera[26] otvorenog koda. Nakon ovoga su kreirane i sintaktičke fraze (*n*-grami) sastavljene od dobijenih POS tagova.

Same reči rečinice takođe predstavljaju dobru indikaciju o njenoj subjektivnosti, odnosno objektivnosti [Turney 2002] pa je kao drugi pogled na rečenicu korišćeno prisustvo određenih reči u rečenici [Slivka 2012]. Kao i u slučaju prvog pogleda, takođe su kreirani *n*-grami sastavljeni od ovih obeležja. Za pretprocesiranje teksta upotrebljen je isti pristup kao onaj opisan u odeljku 4.1.1.

U slučaju prirodne podele obeležja, pogledi bi trebali da budu međusobno nezavisni. POS tagovi reči svakako nisu nezavisni od samih reči. Međutim, kao što će se videti iz eksperimenata prikazanih u ovom odeljku, opisana podela obeležja u kombinaciji sa ko-treningom u značajnoj meri unapređuje performanse inicijalnog klasifikatora. Budući da u datom slučaju raspolažemo isključivo sa rečima teksta, teško je definisati potpuno nezavisne poglede bez upotrebe dodatnih izvora informacija.

Budući da detekcija subjektivnosti predstavlja specijalan slučaj kategorizacije teksta, skupa zadataka koji se odlikuju velikom redudantnošću skupova podataka [Joachims 2001], kao i zbog činjenice da definisani pogledi sadrže potencijalno redudantne informacije, očekujemo dobre performanse ko-trening algoritma sa slučajnom podelom obeležja, a samim tim i ostalih postavki baziranih na njoj (*RSSalg* i MV).

6.1.3 Rezultati i diskusija

U cilju evaluacije predloženog rešenja korišćen je skup podataka *subjectivity dataset v1.0*[27] [Pang 2004]. Da bi se obučio model za detekciju

[26] Stanfordov parser je dostupan na sajtu http://nlp.stanford.edu/software/lex-parser.shtml
[27] Skup podataka je dostupan na sajtu www.cs.cornell.edu/people/pabo/movie-review-data/

subjektivnosti na nivou rečenice, potrebna je velika kolekcija anotiranih rečenica. Međutim, veoma je teško doći do kolekcije individualnih rečenica koje se lako mogu anotirati kao isključivo subjektivne ili objektivne [Riloff 2003b]. Autori u [Pang 2004] ovo rešavaju kreiranjem automatski anotiranog skupa podataka. Za ovaj skup podataka je kolektovano 5000 rečenica iz revizija filmova sa sajta www.rottentomatoes.com i 5000 rečenica iz sumarizacija radnja filmova dostupnih na sajtu www.imdb.com. Rečenice ekstrahovane iz revizija filmova tretirane su kao subjektivne, a rečenice ekstrahovane iz opisa radnje kao objektivne rečenice. Ova pretpostavka o subjektivnosti, odnosno objektivnosti rečenica je uglavnom tačna, mada se među opisima radnje mogu povremeno naći i subjektivne rečenice koje su u ovom korpusu pogrešno anotirane kao objektivne.

Radi evaluacije performansi predloženih rešenja korišćena je stratifikovana desetostruka unakrsna validacija opisana u poglavlju 4. Kao mera performansi poređenih modela modela korišćena je tačnost (*accuracy*). Mere kao što su *f*-mera, preciznost i odziv nisu korišćene u ovom eksperimentu, budući da je skup podataka veoma balansiran – osnovni model (*baseline*) koji sve instance svrstava u najzastupljeniju klasu, ima tačnost 50.0%.

Algoritam mašinskog učenja korišćen u okviru ko-trening algoritma je naivni Bajes (NB), izabran zbog svoje brzine (koja predstavlja važan factor zbog kompleksnosti *RSSalg* postavke), kao i zbog toga što se NB generalno pokazao kao dobar model za probleme kategorizacije teksta [Lewis 1993].

Kao inicijalni anotirani obučavajuči skup L na slučajan način je odabrano 10 subjektivnih i 11 objektivnih rečenica. Skup je odabran tako da inicijalni klasifikator nije suviše slab, ali i da ko-trening ima prostora da unapredi performanse, odnosno da je optimalan dobitak dovoljno velik (10-20%). Tačnost koju postiže NB klasifikator treniran na inicijalnom obučavajučem skupu L je L_{acc} = (58.0 ± 2.2)%, a tačnost koju postiže NB klasifikator treniran na obučavajućem skupu sastavljenom od inicijalnog anotiranog skupa L i neantoriranog skupa U kome su dodeljena tačna klasna obeležja je All_{acc} = (76.3 ± 0.4)%. Tačnost All_{acc} možemo interpretirati kao ciljnu tačnost koju bismo želeli da postignemo primenom polu-nadgledanog obučavanja.

Broj primera označenih od strane unutrašnjih klasifikatora ko-treninga u svakoj iteraciji (brojevi n i p, odeljak 1.3.4) je odabran u skladu sa stvarnom distribucijom originalnog skupa podataka – u svakoj iteraciji se u inicijalni anotirani skup dodaje 5 najpouzdanije anotiranih pozitivnih rečenica i 5 najpouzdanije anotiranih negativnih rečenica. Veličina podskupa neanotiranih primera *u'* (*unlabeled pool*, odeljak 1.3.4) je 50, a broj iteracija ko-trening algoritma je 20 kao i u [Feger 2008], radi što boljeg poređenja sa $MaxInd_{best}$ algoritmom. Broj različitih slučajnih podela korišćenih u *RSSalg* metodologiji

(broj *m*, odeljak 3.1) je 100, što je odabrano u skladu sa rezultatima prikazanim u odeljku 4.5.1.

Prosečna tačnost postignuta od strane poređenih ko-trening algoritama u desetostrukoj unakrsnoj validaciji je zapisana u tabeli 17.

Algoritam	Algacc	Algacc – Lacc	Allacc – Algacc
Random	59.5 ± 4.3	1.5	16.8
Natural	61.0 ± 3.0	3.0	15.3
MaxIndbest	56.4 ± 5.2	-1.6	23.2
MV	67.5 ± 3.1	9.5	8.8
RSSalg	64.9 ± 3.4	6.9	11.4
RSSalgbest	68.4 ± 2.3	10.4	7.9

Tabela 17 Tačnost i standardna devijacija [%] za primenjene ko-trening algoritame. Kolona $\mathbf{Alg_{acc}}$ predstavlja prosečnu tačnost koju je primenjeni algoritam postigao u desetostrukoj unakrsnoj validaciji. Kolona $\mathbf{Alg_{acc} - L_{acc}}$ ukazuje koliko je algoritam pobojšao performanse u odnosu na polazni klasifikator. Negativna vrednost u ovoj koloni ukazuje na to da je došlo do degradacije performansi u odnosu na polazni klasifikator. Kolona $\mathbf{All_{acc} - Alg_{acc}}$ ukazuje na razliku ciljne tačnosti i tačnosti postignute od strane primenjenog algoritma.

Na osnovu rezultata prikazanih u tabeli 17 možemo da vidimo da je *Random* postavka uspela da unapredi performase polaznog klasifikatora. Ovaj rezultat takođe potvrđuje rezultate [Nigam 2000b] – ukoliko postoji dovoljna redudancija u skupu obeležja, ko-trening sa slučajnom podelom obeležja može da unapredi performanse polaznog klasifikatora. $MaxInd_{best}$ postavka je degradirala performanse polaznog klasifikatora. Postavka *Natural* koja tretira postojanje određenih POS tagova reči u rečenici kao jedan od pogleda na rečenicu, a postojanje određenih reči u rečenici kao drugi pogled na rečenicu, se pokazala uspešna u ovom eksperimentu – unapredila je performanse polaznog klasifikatora. Ova postavka je takođe pokazala dva puta veće uvećenje tačnosti polaznog klasifikatora od *Random* postavke.

RSSalg postavka je u ovom eksperimentu uspela da unapredi performanse polaznog klasifikatora, a takođe je pokazala i značajno bolje performanse od $MaxInd_{best}$, *Random* i *Natural* postavke.

Sa druge strane, *RSSalg* postavka u ovom eksperimentu nije uspela da dostigne svoje ciljne performanse ($RSSalg_{best}$ postavku). Budući da je pokazala gore performanse od *MV* postavke, možemo zaključiti da postupak kombinacije predikcija pojedinačnih klasifikatora nije bio uspešan. $RSSalg_{best}$ postavka je pokazala najbolje performanse od svih testiranih postavki. Dakle, sa pravim izborom parametara *RSSalg* postavka bi se pokazala uspešnija od svih alternativa. Možemo zaključiti da je u budućnosti potrebno korigovati proceduru automatske detekcije pragova, ali i da se predloženi model pokazao uspešan na zadatku detekcije subjektivnosti, budući da je u značajnoj meri unapredio performanse polaznog klasifikatora i pokazao bolje performanse od $MaxInd_{best}$, *Random* i *Natural* alternativa.

Potrebno je još prokomentarisati da je u radu [Pang 2008] nadgledanim obučavanjem dobijena tačnost od 92%, što je u značajnoj meri veća tačnost od

one dobijene u ovde izvršenim eksperimentima primenom nadgledanog obučavanja na velikom obučavajućem skupu (76.3%). Međutim, mora se imati u vidu da su autori rada vršili pažljiv odabir obeležja, dok su ovde korišćena ona najosnovnija. Cilj ove ealuacije nije bio da se uvećaju performanse već predloženih modela detekcije subjektivnosti, već da se pronađe način na koji bi se ko-trening mogao primeniti na ovaj problem u cilju olakšavanja problema mukotrpnog i skupog anotiranja velikog broja dokumenata. U budućnosti bi svakako trebalo isprobati i tehnike odabira obeležja koje bi se mogle primeniti u skladu sa ko-trening postavkom radi pobojšanja performansi ovog rešenja.

6.1.4 Zaključak

U ovom odeljku pokazano je da se ko-trening algoritam može uspešno primeniti na problem detekcije subjektivnosti u tekstu u situacijama gde ne raspolažemo sa dovoljno velikim obučavajućim skupom koji bi omogućio uspešnu primenu modela nadgledanog obučavanja. Problem detekcije subjektivnosti je u izloženom eksperimentu posmatran kao klasifikacioni problem, a anotacije subjektivnosti, odnosno objektivnosti teksta su dodeljivane na nivou rečenice.

U izvedenim eksperimentima se *RSSalg* model, predstavljen u ovoj disertaciji, pokazao superiornim u odnosu na *Random*, $MaxInd_{best}$ i *Natural* ko-trening alternative. Međutim, pokazalo se da način odabira parametara ovog modela nije uspešan – po performansama je ovaj model bio gori od većinskog glasanja iste grupe formiranih ko-trening klasifikatora. Sa druge strane, sa pravim izborom pragova *RSSalg* model je nadmašio sve poređene alternative, što znači da je u budućnosti potrebno pobojšati proceduru optimizacije njegovih parametara. Ipak, budući da je *RSSalg* model u značajnoj meri unapredio performanse polaznog klasifikatora, možemo zaključiti da je algoritam uspešno primenjen na zadatak detekcije subjektivnosti.

6.2 Višekategorijska klasifikacija

U cilju automatizacije kategorizacionog zadatka prvo moramo definisati listu postojećih kategorija, a zatim i pripremiti primere već kategorizovanih podataka na osnovu kojih možemo obučiti klasifikator (obučavajući skup). Za obuku klasifikatora visokih performansi potrebno je da obučavajući skup bude što veći i raznovrsniji. Međutim, do anotiranih primera se teško dolazi jer je anotacija dugotrajan i skup proces. Problem nedostatka anotiranih primera je još izraženiji u slučaju razvrstavanja objekata u više katogorija: ručna dodela jedne od više kategorija je izazovniji zadatak od dodele atomičke oznake klase [Guo 2012], a potrebno je anotirati dovoljno raznovrsnih primera za svaku od kategorija. U ovom odeljku biće izvršena empirijska evaluacija primenljivosti ko-treninga na višekategorijske skupove podataka bez prirodne podele obeležja u cilju olakšavanja problema mukotrpne ručne anotacije dokumenata. Rezultati

primene sistema na više-kategorijsku klasifikaciju publikovani su u radovima [Slivka 2011a], [Slivka 2011b] i [Slivka 2012b].

Originalni ko-trening algoritam je dizajniran za binarne klasifikacione probleme. Iako je lako modifikoveti ovaj algoritam da radi sa višekategorijskim skupovima podataka, dosadašnja istraživanja vezana za ko-trening su se uglavnom fokusirala na binarne klasifikacione probleme. Postavlja se pitanje da li se zaključci dobijeni u izvedenim studijama generalizuju na realne klasifikacione probleme sa velikim brojem kategorija [Ghani 2002b]. Razlog zbog koga bi ovo moglo biti problematično jeste potreba za odgovarajućom podelom obeležja, koja zahteva da dva skupa obeležja budu konzistentna u smislu da ciljna funkcija formirana osnovu svakog od pogleda mora da predviđa istu anotaciju za većinu primera. Recimo, predikcija kategorije jedne Web stranice bi trebala da bude ista, bilo da su za predikciju korišćena obeležja nastala na osnovu reči iz linkova koji ukazuju na datu web stranicu ili su za predikciju korišćene reči same Web starnice. Postavlja se pitanje da li se ovaj zahtev može ispuniti u dovoljnoj meri kada u datom obučavajućem skupu postoji veliki broj kategorija [Shinnou 2004].

U praksi se višekategorijski problemi često rešavaju dekompozicijom na višestruke binarne klasifikacione probleme. Nakon toga, ko-trening se može primeniti na individualne binarne probleme. Takođe, mnogi autori su prilikom ispitivanja ko-treninga vršili i konverziju višekategorijskih klasifikacionih problema u jedinstven binarni problem spajanjem više klasa u jednu (pozitivnu klasu) i tretiranjem svih preostalih klasa kao duge (negativne klase). U praksi, ovo može da dovede do zanemarivanja neke od manjih, ali značajnih klasa. Nasuprot tome, u ovde izloženom pristupu se problemom višekategorijske klasifikacije rukuje direktno, bez konverzije višekategorijskog problema u jedan ili više binarnih klasifikacionih problema.

Za eksperimente je korišćeno nekoliko višekategorijskih UCI skupova podataka. Ovi skupovi podataka nemaju prirodnu podelu obeležja (ili, barem, odgovarajuća podela nije poznata). Zbog toga su u eksperimentu primenjene sledeće ko-trening postavke za koje nije potrebno poznavanje prirodne podele obeležja: *Random*, $MaxInd_{best}$, *RSSalg* i *IMCC* (postavke su detaljno opisane u odeljku 4.3). Ove postavke su takođe upoređene sa performansama NB klasifikatora obučenog na malom anotiranom skupu podataka, kao i NB klasifikatora obučenog na značajno većem skupu podataka koji orijentaciono predstavlja ciljne performanse koje želimo da postignemo.

U izvršenim eksperimentima su najbolje performanse pokazali modeli predloženi u ovoj disertaciji – *RSSalg* i *IMCC*. Pokazano je da se na način opisan u ovom odeljku predloženi modeli mogu uspešno primeniti na višekategorijske skupove podataka, ali da je i u ovoj primeni neophodno pronaći bolji način estimacije parametara *RSSalg* postavke.

Ovaj odeljak je organizovan na sledeći način. U odeljku 6.2.1 dat je pregled radova u kojima se ko-trening primenjuje na probleme višekategorijske klasifikacije. U odeljku 6.2.2 izloženo je kako se sistemi predstavljeni u ovoj disertaciji mogu primeniti na višekategorijsku klasifikaciju. U odeljku 6.2.3 izložen je postupak validacije predloženog rešenja, postignuti rezultati i njihova diskusija. Konačno, odeljak 6.2.4 zaključuje ovaj odeljak.

6.2.1 Pregled vladajućih stavova i shvatanja u literaturi

Može se reći da eksperiment predstavljen u ovom odeljku povezuje dva pravca istraživanja – istraživanje mogućnosti primene ko-trening algoritma na skupove podataka bez prirodne podele, kao i istraživanje mogućnosti primene ko-treninga na višekategorijske skupove podataka. Pored pregleda literature već izloženog u poglavlju 2, ovde će biti izloženo još nekoliko radova koji uključuju primenu ko-treninga na višekategorijske skupove podataka.

U radu [Du 2010] predloženo je nekoliko metodologija za kreiranje veštačkih podela obeležja za ko-trening. Takođe je predložena i metodologija verifikacije veštačke podele. Eksperimenti izvršeni u ovom radu obuhvataju i primenu ko-treninga na nekoliko višekategorijskih UCI skupova podataka, međutim, u eksperimentu su autori konvertovali višekategorijski problem u binarni spajanjem više klasa u jedinstvenu klasu. U njihovom eksperimenu je najzastupljenija kasa tretirana kao pozitivna, a sve ostale klase su spojene u jednu (negativnu) klasu.

Sličan postupak u eksperimentima sa višekategorijskim skupovima podataka je izvršen u [Huang 2010]. I ovde su višekategorijski skupovi podataka pretvoreni u binarne spajanjem nekoliko klasa u pozitivnu i nekoliko klasa u negativnu klasu, na taj način da rezultujući skup podataka bude relativno balansiran.

Iako se ovim postupkom mogu demonstritai dobre performanse ko-treninga na binarnim problemima, ne postoji garancija da se zaključci izvedeni na ovako postavljenim eksperimentima generalizuju na slučaj kada postoji više kategorija. Takođe, primenom ovog postupka u realnim problemima bi se zanemarile neke od potencijalno važnih klasa, naročito u relativno izbalansiranim skupovima podataka ili gde je klasa od interesa mala.

Autori u [Ghani 2002b] prezentuju pristup višekategorijskoj klasifikaciji zasnovan na ko-treningu. U njihovom postupku se višekategorijski problem dekomponuje na nekoliko binarnih problema. Dekompozicija na binarne probleme vrši se postupkom izlaznih kodova za korekciju greške (*error-correcting output codes*) [Dietterich 1995]. U ovom postupku se problem od *m* klasa konvertuje u *n* binarnih problema, gde *n* može biti i manje od *m*. Svakoj klasi se dodeljuje jedinstven binarni string dužine *n* koji se naiziva kodna reč (*codeword*). Kao prvi korak, formira se matrica *M* dimenzija $m \times n$. Svaka kolona matrice *M* deli prostor svih klasa na dva dela. Nakon toga se obučava *n*

klasifikatora za svaki od *n* binarnih problema (u postupku opisanom u [Ghani 2002b] se za obučavanje klasifikatora koristi ko-trening). Nakon obuke, klasifikacija nove test instance se vrši tako što se svaki od klasifikatora primenjuje na datu instancu. Predikcije klasifikatora se kombinuju u cilju dobijanja koda dužine *n* tako što svaki od klasifikatora vrši predikciju jednog bita datog koda. Instanci se dodeljuje klasa čija je kodna reč najbliža dobijenom kodu. U eksperimentima izvedenim u [Ghani 2002b] se pokazalo da je ovaj pristup efektivan u slučaju problema tekstualne kategorizacije koja uključuje veliki broj kategorija. Jedna od pretpostavki koje uvodi ovaj pristup je da pojedinačni obučeni ko-trening klasifikatori imaju dobre performanse i vrše kvalitetnu klasifikaciju. U njihovoj postavci je korišćen ko-trening sa slučajnom podelom obeležja koji se pokazao dobar na problemima tekstualne kategorizacije. Međutim, ovaj pristup ne bi bio efikasan na manje redudantnim skupovima podataka gde ko-trening sa slučajnom podelom obeležja nije efektivan. U tom slučaju, morala bi se primeniti neka druga metodologija za primenu ko-treninga na skupove podataka bez prirodne podele. Iako bi se ovaj pristup mogao kombinovati sa *RSSalg* postavkom, zbog vremenske kompleksnosti *RSSalg* postavke ovaj postupak bi bio veoma dugotrajan jer bi smo morali da obučimo po jedan *RSSalg* klasifikator za svaki od *n* klasifikacionih potproblema. Zbog toga smo se u ovoj evaluaciji ipak odlučili za direktnu primenu ko-treninga na višekategorijske probleme.

6.2.2 Višekategorijska ko-trening postavka

U svojoj originalnoj formulaciji, ko-trening algoritam je primenljiv na skupove podataka u kojima postoje samo dve klase – pozitivna i negativna. Međutim, ko-trening je jednostavno primeniti i na višekategorijske skupove podataka time što se za unutrašnje klasifikatore odaberu klasifikacioni modeli koje je moguće primeniti na višekategorijske skupove podataka. U ovom slučaju klasifikatorima se dozvoljava da u svakoj iteraciji, za svaku od klasa skupa podataka, anotiraju predefinisan broj primera za koje je su najpouzdaniji da pripadaju datoj klasi. Ovaj postupak je predstavljen pseudo-kodom u algoritmu 7.

Ulaz

- Skup kategorija $\{C_k\}$ u koje je neophodno razvrstati instance, $k \in \{1..K\}$
- Mali skup L anotiranih instanci
- Znatno veći skup U primera koji nisu anotirani
- Skup neanotiranih instanci T koje je neophodno razvrstati u skup klasa $\{C_k\}$.
- Skup obeležja X kojima su opisani dati skupovi podataka
- Parametri ko-trening algoritma:
  - podela skupa obeležja X na skupove obeležja X_1 i X_2
  - broj iteracija ko-trening algoritma k
  - veličina podskupa neanotiranih primera u'
  - za svaku od klasa $k \in \{1,...,K\}$ broj primera c_k koji će u svakoj iteraciji biti anotirani datom klasom i dodati u inicijalni obučavajući skup)
  - unutrašnji klasifikatori h_1 i h_2 koji podržavaju višekategorijske skupove podataka

Izlaz

- Uvećan obučavajući skup L_{res} koji se sastoji od inicijalnog obučavajućeg skupa L i primera anotiranih i dodatih u skup L u toku ko-trening procesa
- Klasifikatori h_1 i h_2 obučeni na uvećanom obučavajućem skupu L_{res}
- Skup $T = \{(x_i, y_i)\}$, $i = 1..n$ gde x_i predstavlja i-tu instancu ulaznog obučavajućeg skupa D, a y_i estimaciju tačnog klasnog obeležja instance x_i

Algoritam

Obučavanje:

Za svako i, i=1..k:

- Kreirati podskup neanotiranih primera U' slučajnim odabirom u' primera iz skupa U.
- Trenirati klasifikator h_1 korišćenjem obučavajućeg skupa L i skupa osobina X_1.
- Trenirati klasifikator h_2 korišćenjem obučavajućeg skupa L i skupa osobina X_2.
- Za svaku od kategorija $C_j \in \{1,\ldots,K\}$:
  - Dozvoliti klasifikatoru h_1 da anotira c_k primera skupa U' za koje je klasifikator najsigurniji da pripadaju klasi C_j.
  - Dozvoliti klasifikatoru h_2 da anotira c_j primera skupa U' za koje je klasifikator najsigurniji da pripadaju klasi C_j.
- Dodati ovako anotirane primere u obučavajući skup L.
- Na slučajan način odabrati $2 \cdot \sum_{j=1}^{K} c_j$ primera iz skupa U i prebaciti ih u skup U'.

Klasifikacija novih primera:

Za svaku instancu $t \in T$ verovatnoća da ta instanca pripada kategoriji $C_j \in \{1,\ldots,K\}$ se računa tako što se pomnože verovatnoće kojom klasifikatori h_1 i h_2 predviđaju da ta instanca pripada kategoriji C_j. Instanci se dodeljuje kategorija kojoj odgovara najveća verovatnoća.

Algoritam 7: Ko-trening algoritam primenjen na višekategorijske skupove podataka

6.2.3 Rezultati i diskusija

Za evaluaciju je odabrano osam UCI višekategorijskih skupova podataka (tabela 18). Ovi skupovi podataka su odabrani budući da su već korišćeni u ko-

trening evaluaciji [Du 2010; Huang 2010]. Osnovne karakteristike skupova podataka su izlistane u tabeli 18.

Skup podataka	Dim	|L|	L_{acc}	|All|	All_{acc}	Optimlan dobitak	Cat
Splice	62	24	66.2	1914	95.3	29.1	3
Wine	14	3	71.9	107	96.2	24.3	3
OptDigits	65	60	72.6	3372	91.7	19.1	10
SyntheticControl	62	24	76.8	360	94.1	17.3	6
Waveform 5000	41	15	64.1	3000	80.0	15.9	3
Dermatology	35	19	81.8	219	97.1	15.3	6
Segment	20	21	67.2	1386	80.8	13.6	7
CMC	10	5	41.1	884	48.8	7.7	3

Tabela 18 Sumarizacija najvažnijih osobina skupova podataka korišćenih za evaluaciju. Notacija: **Dim** – dimenzionalnost, odnosno, broj instanci skupa podataka; **|L|** - veličina malog anotiranog skupa podataka u formatu „broj instanci koje pripadaju pozitivnoj klasi/broj instanci koje pripadaju negativnoj klasi"; $\mathbf{L_{acc}}$ – tačnost koju postiže NB klasifikator obučen na malom skupu anotiranih podataka *L*; **|All|** - veličinacelog obučavajućeg skupa (t.j. zbir borojeva instanci malog anotiranog skupa *L* i neanotiranog skupa *U*), takođe u formatu „broj instanci koje pripadaju pozitivnoj klasi/broj instanci koje pripadaju negativnoj klasi"; $\mathbf{All_{acc}}$ –tačnost koju postiže NB klasifikator obučen na celom obučavajućem skupu *All* (t.j. na obučavajućem skupu koji se sastoji od anotiranih instanci skupa *L* i instanci neanotiranog skupa *U* kojima je dodeljena tačna anotacija); **Optimalan dobitak** – procena mogućeg pobojšanja tačnosti u odnosu na polaznu tačnost L_{acc}. Računa se po formuli $All_{acc} – L_{acc}$; **Cat** – broj kategorija skupa podataka.

Eksperiment je izvršen primenom unakrsne validacione procedure opisane u poglavlju 4. Kao mera performansi rešenja korišćena je tačnost (*accuracy*). Skupovi predstavljeni u tabeli 18 nisu u značajnoj meri neizbalansirani, tako da za evaluaciju perfromansi nisu korišćene druge mere kao što su *f*-mera, preciznost i odziv.

U eksperimentu su korišćene sledeće postavke (predstavljene u odeljku 4.3): *Random*, $MaxInd_{best}$, *RSSalg*, $RSSalg_{best}$ i *IMCC*. Parametri ovih postavki odabrani su na isti način kao što je opisano u odeljku 4.2. Prosečna tačnost svake od testiranih postavki dobijena u proceduri unakrsne validacije je predstavljena u tabeli 19.

Datasets	Random	MV	$MaxInd_{best}$	RSSalg	RSSalg best	IMCC
Splice	81.1±6.9	84.1±3.0	77.3±8.5	84.3±3.2	86.2±3.2	**93.1±0.3**
Wine	92.5±6.8	94.5±2.6	92.3±3.5	92.0±4.3	96.8±1.9	**97.8±0.9**
OptDigits	77.4±3.2	82.3±2.0	**88.3±1.7**	83.4±1.7	83.4±1.7	87.9±0.3
SyntheticControl	84.5±4.1	85.0±2.3	87.8±4.1	85.2±2.7	**87.9±2.8**	86.7±1.1
Waveform5000	63.0±7.6	67.1±6.8	64.0±4.4	63.6±2.9	72.0±6.1	**79.8±0.6**
Dermatology	86.9±4.3	87.1±4.2	83.6±2.2	84.2±2.8	87.6±3.9	**97.3±1.7**
Segment	59.6±4.7	63.2±4.1	62.6±3.9	65.4±4.1	72.6±2.9	**76.6±1.7**
CMC	37.3±3.5	38.2±3.1	38.6±3.0	38.1±2.5	45.0±3.3	**47.5±1.4**

Tabela 19 Poređenje alternativnih ko-trening postavki. U tabeli je prikazana postignuta tačnost i standardna devijacija (u procentima) dobijena u proceduri stratifikovane 10-struke unakrsne validacije na svakom od skupova podataka. Najveća postignuta tačnost za svaki od skupova podataka je označena masnim slovima (*bold*).

Na osnovu rezultata prikazanih u tabeli 19 najbolje performanse na višekategorijskim skupovima podataka je pokazala *IMCC* postavka. Ova postavka je na svim skupovima podataka unapredila polazni klasifikator, a na većini skupova podataka njene performanse su bliske performansama *All* postavke. Nakon *IMCC* postavke, najbolje performanse je pokazala $RSSalg_{best}$ postavka koja predstavlja *RSSalg* postavku sa idealno odabranim parametrima. Ova postavka je takođe na svim skupovima podataka uspela da unapredi polazni klasifikator. *RSSalg* postavka većinom nije uspela da dostigne performanse $RSSalg_{best}$ postavke. Takođe, performanse ove postavke su veoma bliske performansama *MV* i $MaxInd_{best}$ postavki. Na CMC i Segment skupovima podataka, *MV* i $MaxInd_{best}$ i *RSSalg* postavka čak degradiraju performanse polaznog klasifikatora. Razlog za loše performanse *MV* i *RSSalg* postavke je u slabim performansama *Random* postavke na koju se oslanjaju, a koja na ovim skupovima podataka takođe ne prikazuje dobre performanse. Najverovatniji razlog za uspeh *IMCC* postavke u ovom slučaju jeste njena veća robustnost na kombinaciju jačine pojedinačih klasifikatora i njihove različitosti (odeljak 4.5.3) u odnosu na ostale testirane postavke, ali detaljna analiza ovog fenomena zahteva dodatne eksperimente i predstavlja temu budućeg rada.

6.2.4 Zaključak

U ovom odeljku ispitivana je mogućnost primene modela predloženih u ovoj disertaciji na višekategorijske skupove podataka. Višekategorijskim problemom je ovde rukovano direktno, malom modifikacijom ko-trening algoritma, nasuprot drugim rešenjima koja konvertuju višekategorijski klasifikacioni problem u više binarnih problema, ili pak u jedan binarni klasifikacioni problem spajanjem više različitih klasa.

Eksperimenti su izvedeni na osam UCI višekategorijskih skupova podataka bez prirodne podele obeležja, korišćenjem procedure unakrsne evaluacije.

U izvedenim eksperimentima najbolje su se pokazale *IMCC* i $RSSalg_{best}$ postavke koje su na svakom skupu podataka unapredile performanse polaznog klasifikatora. Ni jednoj drugoj postavci ovo nije uspelo u tolikoj meri. *IMCC* postavka se pokazala nešto boljom od $RSSalg_{best}$ postavke, a njene performanse su u rangu performansi NB klasifikatora obučenog na značajno većem obučavajučem skupu. Performanse *RSSalg*, *MV* i $MaxInd_{best}$ postavke su u ovim eksperimentima bile veoma slične.

Rezultati pokazuju da se na način opisan u ovom odeljku modeli prikazani u ovoj disertaciji mogu uspešno primeniti na višekategorijske skupove podataka, ali takođe ukazuju na potrebu za boljim načinom estimacije parametara *RSSalg* postavke.

6.3 Sistemi za davanje preporuka

U današnje vreme, korisnici su preplavljeni ogromnom količinom raspoloživih izbora prilikom kupovine, gledanja filmova, traženja restorana, potrage za obrazovanjem, itd. Zbog toga se mnogi web sajtovi oslanjaju na sisteme za davanje preporuka (*recommender systems*) zarad personalizacije željenog sadržaja za datu mušteriju [Resnick 1997]. *Amazon*[28] koristi sisteme za davanje preporuka u cilju personalizacije preporuka proizvoda za kupce [Linden 2003]. *MovieLens*[29] nudi personalizovane preporuke za filmove koje korisnik nije ocenio [Chen 2010]. *TripAdvisor* je web sajt koji asistira korisnike u njihovoj potrazi za informacijama o putovanju[30] [Wang 2012].

Pristupi za izgradnju sistema za davanje preporuka bi se mogli razvrstati u 3 pravca – kolaborativno filtriranje (*collaborative filtering*), filtriranje sadržaja (*content-based filtering*) i hibridni pristup (*hybrid filtering*) koji kombinuje dva prethodno navedena pristupa. Algoritmi za kolaborativno filtriranje su bazirani na sličnosti korisnika. Ovi algoritmi uvode pretpostavku da će korisnici koji imaju sličan ukus davati slične ocene istim artikalima. Sa druge strane, algoritmi za filtriranje sadržaja, za datog korisnika, predikciju rejtinga za određeni artikal daju na osnovu istorije rejtinga i relevantnosti opisa sadržaja artikala za datog korisnika. Oba navedena pristupa prilikom obuke modela za predikciju budućih rejtinga korisnika zahtevaju poznavanje istorije rejtinga datog korisnika i nedostatak ovih informacija može u značajnoj meri degradirati performanse sistema za preporuku. Problem nedostatka istorije rejtinga za datog korisnika se javlja prilikom pristupa novog korisnika sistemu i poznat je kao problem hladnog starta kod novog korisnika (*new-user cold-start problem*) [Adomavicius 2005].

U ovom odeljku biće opisan način da se dati problem hladnog starta kod novog korisnika ublaži time što će se smanjiti količina rejtinga koju istorija datog korisnika mora da sadrži da bi se mogao izgraditi precizan model za davanje preporuka. Drugim rečima, za novog korisnika koji je ocenio veoma malo artikala, cilj je da se izgradi sistem za davanje preporuka koji će imati iste performanse kakve bi imao ukoliko bi mu bila dostupna znatna količina rejtinga datog korisnika. Rezultati primene sistema predstavljenih u ovoj disertaciji na problem hladnog starta kod pojave novog korisnika u sistemu za davanje preporuka publikovani su u radu [Slivka 2012a].

U odeljku 6.3.1 biće dat pregled literature u kojoj se ko-trening predlaže kao rešenje problema vezanih za obuku sistema za davanje preporuka. U odeljku 6.3.2 biće izložen detaljan pregled ovde primenjene metodologije za rešavanje problema pojave novog korisnika prilikom obuke sistema za davanje preporuka.

[28] http://www.amazon.com/
[29] http://www.movielens.org/
[30] www.tripadvisor.com

Odeljak 6.3.3 daće pregled i diskusiju postignutih rezultata, a odeljak 6.3.4 će prikazati izvedene zaključke i navesti moguće pravce daljeg istraživanja.

6.3.1 Pregled vladajućih stavova i shvatanja u literaturi

Autori u [Billsus 1998] predlažu sistem za davanje preporuka baziran na kolaborativnom filtriranju. U ovoj postavci, kolaborativno filtriranje je tretirano kao problem klasifikacije i shodno tome, ocene korisnika su diskretizovane u mali broj klasa. Za svakog korisnika se gradi poseban model koji za dati artikal vrši predikciju kojoj od klasa (definisanih na osnovu mogućih ocena) dati artikal pripada. Obučavajući skup na osnovu koga se gradi model za datog korisnika je oformljen na taj način da se kolaborativno filtriranje svodi na primenu odabranog algoritma mašinskog učenja za obučavanje klasifikatora. Ovde je usvojen ovaj postupak u cilju primene klasičnog ko-trening algoritma koji je originalno definisan za klasifikacione probleme.

U [Qu 2013] razvijen je sistem za preporuku filmova baziran na filtriranju sadržaja. Ovaj sistem se bazira na integraciji sadržaja tri različita izvora informacija koja opisuju film: slike, teksta i zvuka. U datoj postavci, svaki od izvora informacija (tip medija) je tretiran kao jedan od pogleda (skupa obeležja) koji opisuju podatke (film). Korišćenjem ovih pogleda, primenjen je ko-trening algoritam u cilju bogaćenja profila korisnika koji su ocenili veoma malo filmova. Slično njihovom radu, ovde je primenjen ko-trening sa istim ciljem ublažavanja problema izgradnje sistema za davanje preporuka za nove korisnike, međutim, postoji nekoliko bitnih razlika ovih rešenja:

- U [Qu 2013] je predložen sistem za preporuke baziran isključivo na filtriranju sadržaja, dok ovde predložen hibridni sistem osim sadržaja artikla koristi i istorije ocena drugih korisnika.
- Takođe, u [Qu 2013] preporuke pojedinačnih pogleda se formiraju na sledeći način: kao prvi korak se pronalazi se *k* najbližih suseda datog filma (*k nearest neighbors*), a ocena za datog korisnika se dodeljuje bazirano na najfrekventnijoj oceni iz skupa *k* najbližih suseda. Ovoj oceni se dodeljuje i ocena predikcije (*score*) bazirano na kombinaciji rastojanja datog filma od njemu najbližih suseda i toga koliko puta je data ocena dodeljena njegovim najbližim susedima. Nasuprot ovom, u ovde su na osnovu opisa artikala i ocene korisnika konstruisana obeležja i primenjen je algoritam mašinskog obučavanja koji na osnovu konstruisanih obeležja trenira klasifikacioni model. Ovakva metodologija može biti primenjena korišćenjem bilo kog klasifikacionog modela.

U radu [Delgado 1999] je predložena ideja da se kreira hibridni sistem za davanje preporuka koji bi, kao i sistem predložen u [Slivka 2014], kao prvi pogled koristio istoriju rejtinga korisnika, a kao drugi pogled koristio opis artikala. Međutim, u ovom radu nisu dati detalji konkretne metodologije kojom bi se ovo postiglo, niti eksperimentalni rezultati.

Još jedan hibridni sistem za davanje preporuka potpomognut ko-trening algoritmom je predstavljen u radu [Ghani 2002a]. Ovde su opisi artikala i ponašanje korisnika analizirani u cilju automatske ekstrakcije semantičkih atributa. Proces ekstrakcije semantičkih atributa je potpomognut ko-treningom. Nakon ekstrakcije, NB klasifikator je primenjen u cilju obuke sistema za preporuku baziranog na filtriranju sadržaja. Za razliku od [Ghani 2002a], u [Slivka 2014] je ko-trening primenjen direktno u kontekstu obuke sistema za preporuku. Ova dva sistema nisu suprotstavljena – ko-trening process predstavljen u [Ghani 2002a] bi se mogao koristiti i u okviru sistema predstavljenog u [Slivka 2014] radi ekstrakcije značajno boljih obeležja iz opisa artikla.

6.3.2 Metodologija

Prilikom pristupa novog korisnika sistemu za davanje preporuka raspoloživ nam je veoma mali obučavajući skup (mali broj artikala ocenjenih od strane datog korisnika), ali, takođe nam je dostupan ogroman neanotirani skup u vidu artikala koje dati korisnik nije ocenio. Ovo je idealna postavka za primenu tehnika polu-nadgledanog obučavanja. U ovom odeljku biće opisano kako se ko-trening može primeniti na zadatu postavku u cilju podizanja performansi sistema za davanje preporuka u opisanom slučaju nedostatka istorije rejtinga.

Ko-trening podrazumeva da se skup obeležja svake instance može podeliti na dva odvojena skupa (pogleda). Kao što je predloženo u [Delgado 1999], dva pogleda koja se koriste u ovde opisanoj postavci za opis podataka su:

- Obeležja bazirana na rejtinzima drugih korisnika, koja se koriste za konstrukciju prediktora baziranog na kolaborativnom filtriranju [Slivka 2012a];
- Obeležja bazirana na opisu artikala, koja se koriste za konstrukciju prediktora baziranog na filtriranju sadržaja [Slivka 2012a].

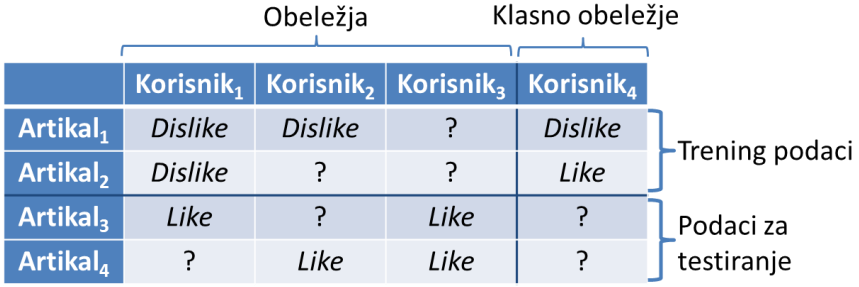
Budući da definisani pogledi predstavljaju dva različita izvora informacija, datu podelu obeležja možemo okarakterisati kao prirodnu.

U cilju primene klasičnog ko-trening algoritma [Blum 1998], problem preporuke artikala je postavljen kao problem klasifikacije. Kao u [Bilsus 1998], za svakog korisnika se trenira poseban klasifikator koga je moguće primeniti na artikale koji korisnik nije ocenio radi predviđanja da li će se artkal dopasti korisniku, te ga je neophodno preporučiti (klasa označena sa „*like*“) ili se artikal neće dopasti korisniku, te ga njemu ne treba preporučivati (klasa označena sa „*dislike*“). Diskretizacija rejtinga korisnika u ove dve klase se vrši zadavanjem praga t i zamenom rejtinga koji prelaze ovaj prag klasom „*like*“, a rejtinga koji se nalaze ispod zadatog praga klasom „*dislike*“.

U odeljcima 6.3.2.1 i 6.3.2.2 je opisan način na koji se konstruišu prediktori bazirani na svakom od pogleda ponaosob, a odeljak 6.3.2.3 opisuje primenjene ko-trening postavke.

6.3.2.1 Prvi pogled: prediktor baziran na kolaborativnom filtriranju

Podaci o rejtinzima korisnika se mogu predstaviti kao retka matrica čiji redovi (instance trening skupa) predstavljaju artikale, a kolone (obeležja) odgovaraju rejtinzima koji su korisnici dodelili datim artikalima [Bilsus 1998]. Ovu matricu nazivamo retkom zbog toga što joj većina vrednosti nedostaje (tipično, svaki korisnik ocenjuje mali podskup svih raspoloživih artikala). U datoj matrici, vrednost obeležja *u* za trening instancu *i* odgovara rejtingu koji je korisnik *U* dao artikalu *I*. Pogled koji je konstruisan na ovaj način zvaćemo *korisnički pogled*. Zadatak predikcije se može posmatrati kao popunjavanje nedostajućih vrednosti u matrici. Za svakog korisnika se trenira poseban model na taj način što se odgovarajuće korisničko obeležje tretira kao klasno obeležje. Za svakog korisnika, ocenjeni artikali se koriste kao anotirani podaci za treniranje modela. Preostali artikali (koji nisu ocenjeni od strane korisnika) se tretiraju kao instance za koje je neophodno odrediti klasno obeležje. Primer ovakve reprezentacije podataka je dat na slici 14. Slika 14 prikazuje izgradnju modela za $Korisnika_4$, u cilju određivanja nedostajućih vrednosti za dato obeležje (ocena koje bi $Korisnik_4$ dodelio datim artiklima). Upitnici u prikazanoj matrici označavaju nedostajuće vrednosti (nedostajuće rejtinge). $Korisnik_4$ je ocenio artikle 1 i 2, zbog čega ove artikle koristimo kao obučavajući skup. $Korisnik_4$ nije ocenio artikle 3 i 4, zbog čega ova dva artikla tretiramo kao instance za koje je neophodno izvršiti predikciju klasnog obeležja (podaci za testiranje, odnosno primenu klasifikatora).



Slika 14 Reprezentacija korišćena za konstrukciju prediktora baziranog na kolaborativnom filtriranju.

Radi treniranja modela za datog korisnika primenjuje se algoritam mašinskog učenja koji može da toleriše nedostajuće vrednosti[31]. Međutim, za nove korisnike koji su ocenili svega nekoliko artikala, rezultujući model bi imao veoma slabe performanse zbog nedovoljno velikog obučavajućeg skupa.

[31] U [Billsus 1998] je opisan i način reprezentacije podataka koji se može koristiti u slučaju kada je potrebno primeniti algoritam mašinskog učenja koji ne toleriše nedostajuće vrednosti.

6.3.2.2 Drugi pogled: prediktor baziran na filtriranju sadržaja

Drugi pogled se sastoji od obeležja kreiranih na osnovu opisa artikla. U ovde izvršenim eksperimentima, kao opis artikala u kontekstu filmova, koriste se obeležja kreirana na osnovu tekstualnog opisa filma (u daljem tekstu ovaj pogled će biti označen kao *pogled radnje*), lista žanrova kojima film pripada (u daljem tekstu *žanr pogled*) i unija ova dva skupa osobina (u daljem tekstu pogled *žanr_radnja*).

Obeležja koja pripadaju *žanr pogledu* se kreiraju tako što se svaki žanr koji postoji u skupu podataka predstavi kao binominalno obeležje sa vrednostima *tačno* (film pripada datom žanru) ili *netačno* (film ne pripada datom žanru).

6.3.2.3 Primenjene ko-trening postavke

U ovde predstavljenim eksperimentima definisano je nekoliko „prirodnih" podela obeležja sa kojima bi se mogao primeniti ko-trening algoritam:

- *Korisnik_radnja*: u ovaj podeli se kao prvi pogled koristi *korisnički* pogled, a kao drugi pogled koristi se pogled *radnje*;
- *Korisnik_žanr*: u ovoj podeli se kao prvi pogled koristi *korisnički* pogled, a kao drugi pogled koristi se *žanr* pogled;
- *Korisnik_žanr_radnja:* u ovoj podeli se kao prvi pogled koristi *korisnički pogled*, a kao drugi pogled se koristi pogled *žanr_radnja*.
- *Žanr_radnja*: u ovoj podeli se kao prvi pogled koristi *žanr* pogled, a kao drugi pogled se koristi pogled *radnje*.

U eksperimentima se koriste sledeće ko-trening postavke definisane u odeljku 4.3: *Natural* (kao prirodne podela obeležja se koriste iznad definisane podele), *Random*, *MV* i *RSSalg*.

6.3.3 Rezultati i diskusija

Radi evaluacije predloženog rešenja razmatra se problem preporuke filmova na popularnom *MovieLens* korpusu[32]. Ovde je korišćen podskup *MovieLens* korpusa za koji je dostupan opis sadržaja filmova skinut sa *IMDB* web sajta[33] [Jannach 2013].

U cilju kreiranja obeležja pogleda *radnje*, korišćene su sledeće tehnike pretprocesiranja teksta: konverzija teksta u mala slova, tokenizator teksta (*string tokenization*), uklanjanje 319 čestih reči engleskog jezika primenom tehnike uklanjanja stop reči i korenovanje reči, izvedeno pomoću Porterovog algoritma [Porter 1980]. Na osnovu tokena dobijenih na opisan način, izgrađen je skup

[32] http://files.grouplens.org/papers/ml-10m.zip
[33] http://www.imdb.com/

podataka baziran na modelu vreće reči uz korišćenje *tf-idf* mere kao vrednosti dobijenih obeležja (odeljak 4.1.1). Filmovima u *MovieLens* skupu podataka korisnici dodeljuju ocene na skali od 1 do 5. Diskretizacija ovih rejtinga je izvršena na sledeći način: {1,2,3}→*Dislike*, {4,5}→*Like*.

Radi evaluacije performansi predloženih rešenja korišćena je stratifikovana desetostruka unakrsna evaluacija opisana u poglavlju 4. Algoritam mašinskog učenja korišćen u okviru ko-trening algoritma je Naivni Bajes (NB), izabran zbog svoje brzine, kao i mogućnosti rada sa nedostajućim vrednostima.

Problem predikcije ocene filmova može biti veoma nebalansiran – pre gledanja filma, korisnici obično konsultuju opis radnje, producenta, glumce i ostale faktore koje smatraju važnima za kvalitet filma, kako bi naslutili da li će im se film dopasti [Qu 2013]. Posledica ovoga je da korisnici generalno gledaju i ocenjuju filmove koji im se dopadaju. Zato, su kao mere performansi modela, pored tačnosti , korišćene i mikro i makro *F*-mera (*micro*/*macro f-measure*) [Sokolova 2009] budući da kod neizbalansiranih skupova podataka pobojšanje tačnosti ne mora uvek da ukazuje na pobojšanje performansi. Neka *tp* i *tn* predstavljaju broj primera ispravno klasifikovanih kao pozitivnih, odnosno negativnih, respektivno, a *fp* i *fn* predstavljaju broj primera pogrešno klasikikovanih kao pozitivnih, odnosno negativnih, respektivno i neka *l* predstavlja broj kategorija. Mikro *f*-mera F_μ se računa prema sledećoj formuli:

$$\begin{aligned} Precision_\mu &= \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l}(tp_i + fp_i)} \\ Recall_\mu &= \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l}(tp_i + fn_i)} \\ F_\mu &= \frac{Precision_\mu \cdot Recall_\mu}{2(Precision_\mu + Recall_\mu)}. \end{aligned} \tag{67}$$

Makro *f*-mera F_M se računa po sledećoj formuli:

$$\begin{aligned} Precision_M &= \frac{\sum_{i=1}^{l} \frac{tp_i}{(tp_i + fp_i)}}{l} \\ Recall_M &= \frac{\sum_{i=1}^{l} \frac{tp_i}{(tp_i + fn_i)}}{l} \\ F_M &= \frac{Precision_M \cdot Recall_M}{2(Precision_M + Recall_M)}. \end{aligned} \tag{68}$$

U eksperimentima se pretpostavlja sledeći scenario: novi korisnik pristupa sistemu u kome već postoji određeni broj korisnika. Novi korisnik je ocenio veoma malo filmova – u našem scenariju novi korisnici ocenjuju 3 filma koja im se svidjaju (pridružuju im klasno obeležje „*like*“) i 3 filma koja im se ne

sviđaju (pridružuju im klasno obeležje „*dislike*“), odnosno, mali inicijalni skup anotiranih primera *L* se sastoji od 3 pozitivna i 3 negativna primera (preostale ocene razmatranih novih korisnika se, ukoliko postoje, ignorišu).

Broj primera označenih od strane unutrašnjih klasifikatora ko-treninga u svakoj iteraciji (brojevi *n* i *p*, odeljak 1.3.4) je odabran tako da se očuvava stvarna distribucija originalnog skupa podataka, kao što je predloženo u [Blum 1998]. Veličina podskupa neanotiranih primera *u'* (*unlabeled pool*, odeljak 1.3.4) je 50 kao u [Feger 2008], a ko-trening algoritam je pušten da iterira sve dok svi primeri iz skupa neanotiranih podataka *U* nisu anotirani. Broj različitih slučajnih podela korišćenih u *RSSalg* metodologiji (broj *m*, odeljak 3.1) je 100, što je odabrano u skladu sa rezultatima prikazanim u odeljku 4.5.1.

U idealnom slučaju, za formiranje *korisničkog* podgleda, želeli bi smo da koristimo sve korisnike sistema uz primenu nekog metoda za redukciju dimenzije problema, ili da odaberemo podskup korisnika koji je najsnažnije koreliran (pozitivno ili negativno) sa korisnikom za koga konstruišemo model. Međutim, zbog malog broja ocena novog korisnika, nemoguće je definisati pouzdanu meru sličnosti kojom bi smo mogli odrediti najsnažnije korelirane korisnike. Zbog toga je, u ovde prikazanim inicijalnim eksperimentima, u cilju formiranja *korisničkog* pogleda korišćeno samo 50 korisnika sa najvećim brojem ocenjenih filmova, budući da će tako formirana matrica biti manje retka.

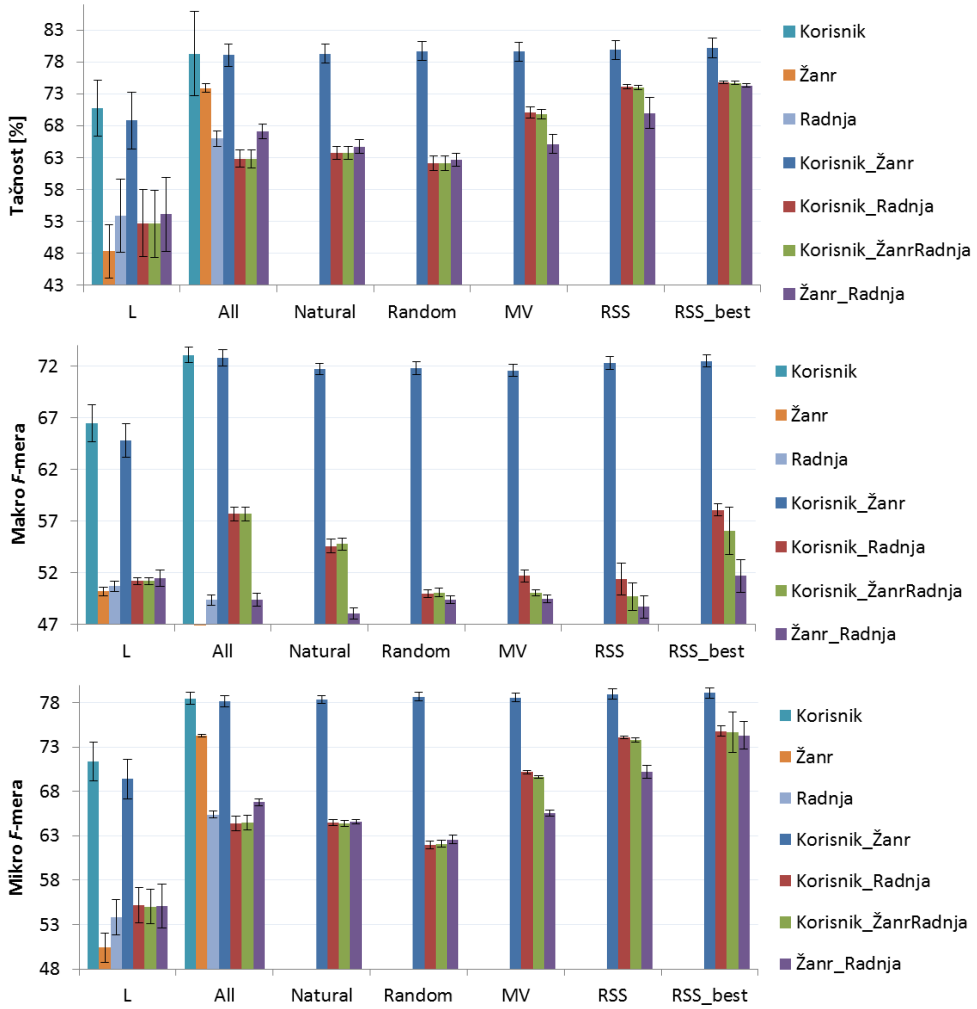
Za eksperiment je na slučajan način odabrano 4 korisnika koji će biti tretirani kao novi korisnici. Prva dva korisnika ($Korisnik_1$ i $Korisnik_2$) pripadaju grupi od 50 korisnika sa najvećim brojem ocenjenih filmova. Druga dva korisnika ($Korisnik_3$ i $Korisnik_4$) ne pripadaju ovoj grupi. Razlog odabira korisnika dve različite grupe je da se osiguramo da ne dobijamo dobre rezultate isključivo iz razoga što su odabrani korisnici visoko korelirani sa 50 korisnika korišćenih u *korisničkom* pogledu. Naime, korišćenjem svih raspoloživih ocena, za odabrane (nove) korisnike su pronađene grupe od 50 njima najsličnijih korisnika. Za korisnike 1 i 2 se ispostavilo da skup njima najsličnijih korisnika prilično preklapa sa skupom od 50 korisnika odabranih za kreiranje *korisničkog* pogleda (preko 25 korisnika ove dve grupe se preklapaju). Međutim, korisnici 3 i 4 imaju samo po jednog korisnika koji pripada i grupi njima najsličnijih korisnika i grupi od 50 korisnika korišćenih za *korisnički* pogled. Grafikoni 24 – 27 (koji sumiraju rezultate prikazane u tabelama) prikazuju postignutu tačnost, mikro i makro *F*-meru za korisnike 1-4, respektivno. Takođe, u tabeli 20 su za $korisnika_1$ dati detalji o tačnosti postignutoj od strane različitih algoritama za različite podele obeležja. Detalji o makro i mikro *f*-meri postignutoj za $korisnika_1$, kao i detalji o postignutoj tačnosti i *f*-merama su izostavljeni, budući da se iz njih mogu izvući identični zaključci. Postavke koje se koriste u eksperimentima (horizontalne ose na grafikonima 24 – 27, odnosno redovi u tabeli 20) su:

- Postavke koje ne koriste podelu obeležja (kombinacije obeležja različitih pogleda se tretiraju kao jedinstveni skup obeležja):
  - *L*: NB klasifikator treniran na skupu anotiranih podataka (6 primera);
  - *All*: NB klasifikator treniran na anotiranim podacima i na neanotiranim podacima kojima je za potrebe ovog eksperimenta dodeljena tačna klasna oznaka. Ovo predstavlja ciljne performanse koje želimo da postignemo primenom ko-trening algoritma (performanse koje bi smo mogli postići nadgledanim obučavanjem ukoliko bi bio raspoloživ veliki broj anotiranih primera).
- Postavke bazirane na podeli obeležja: *Natural, Random, MV, RSS* i *RSS_best*, predstavljene u odeljku 4.3.

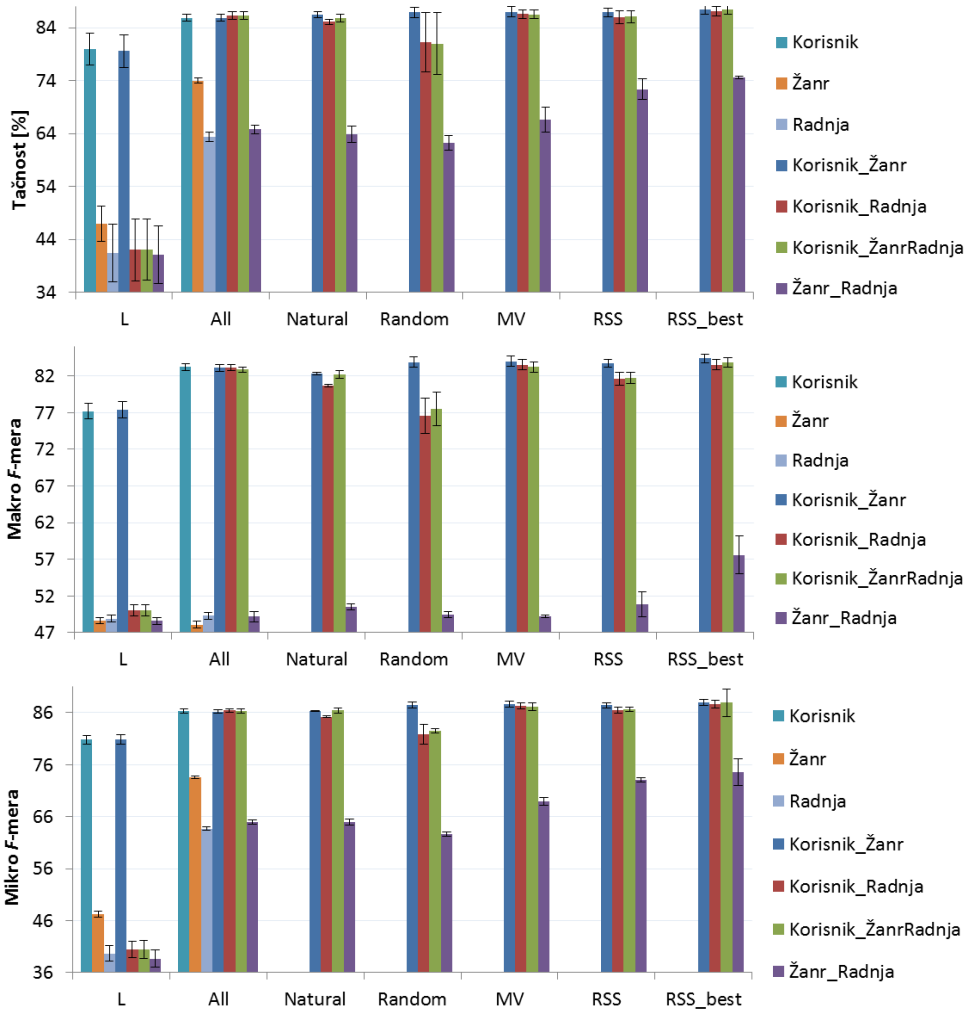
Navedene postavke su testirane korišćenjem različitih pogleda koji su predstavljeni u odeljcima 6.3.2.1 do 6.3.2.3. Korišćeni pogledi su predstavljeni kolonama u tabeli 20.

	Korisnik	***Žanr***	***Radnja***	***Korisnik _Žanr***	***Korisnik _Radnja***	***Korisnik_ ŽanrRadnja***	***Žanr _Radnja***
L	74.7±8.2	48.3±8.4	53.9±11.5	74.5±8.1	62.2±15.7	62.1±15.7	54.1±11.6
All	79.9±3.2	73.9±1.3	66.0±2.4	79.7±3.2	79.9±2.8	79.9±3.0	67.1±2.2
Natur.				80.7±2.9	72.3±16.4	80.4±3.2	64.7±2.2
Rand.				80.2±3.7	71.3±16.7	71.3±16.9	62.7±2.0
MV				80.4±3.1	76.4±11.2	75.8±12.7	65.1±3.0
RSS				80.3±3.3	80.0±4.2	79.8±3.8	70.0±4.8
RSS best				80.9±3.2	81.0±3.9	81.0±3.8	74.3±0.5

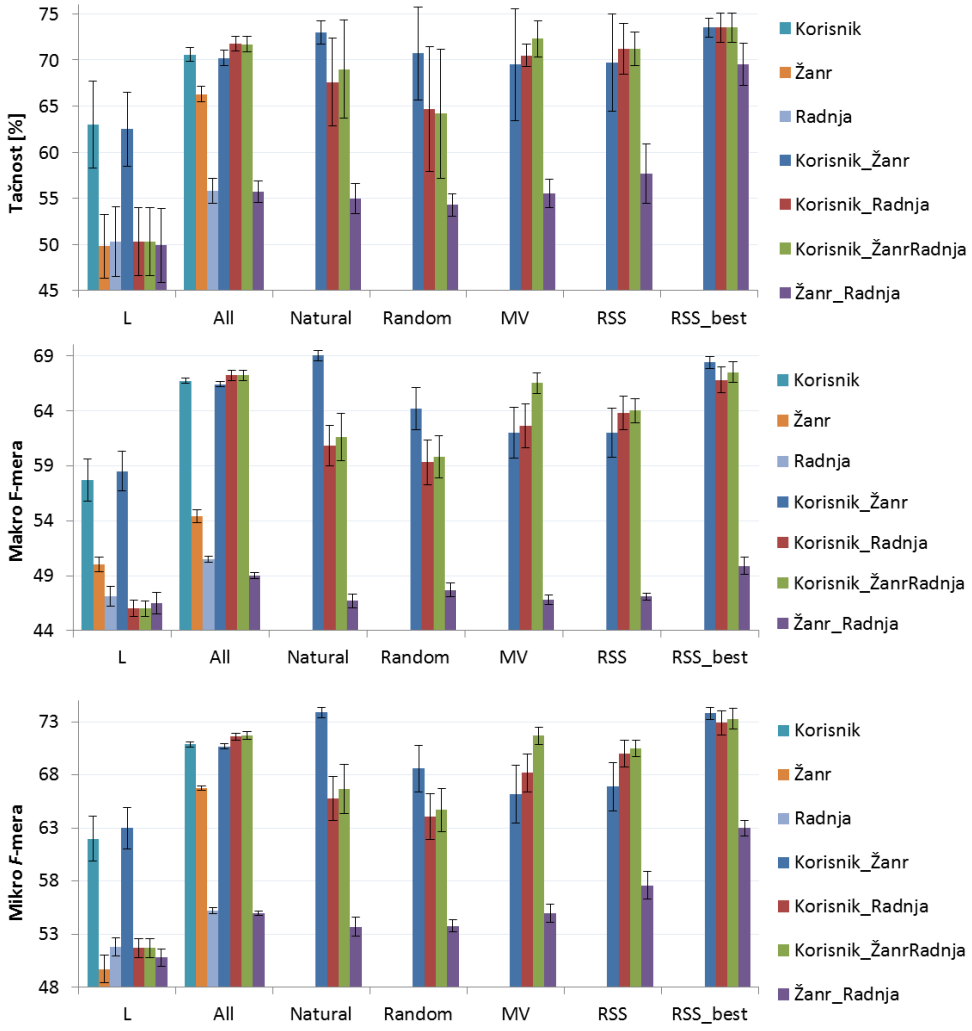
Tabela 20 Korisnik$_1$: Tačnost i standardna devijacija za različite kombinacije algoritama i korišćenih podela obeležja. Broj anotiranih primera korišćenih u *All* postavci je 541, a broj anotiranih primera korišćenih u *L* postavci i ko-trening postavkama je 6.



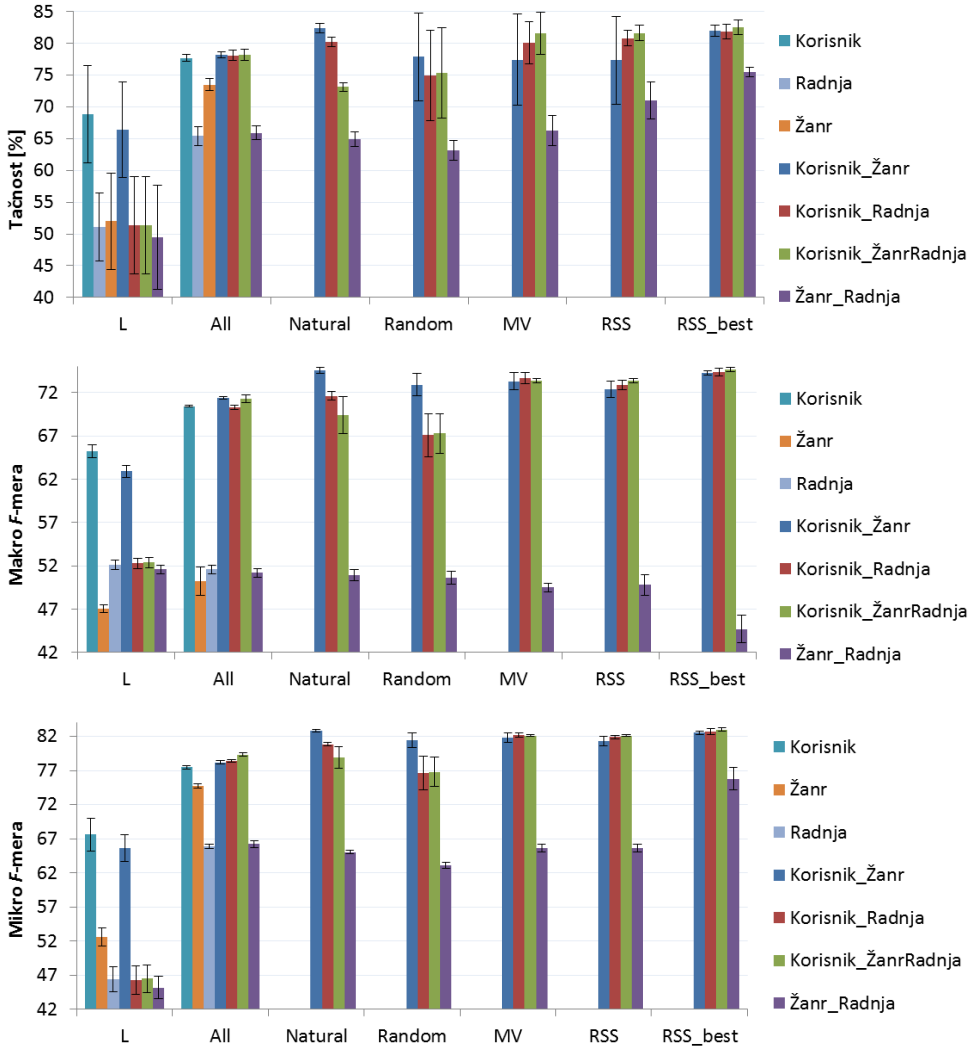
Grafikon 24 *Korisnik*$_1$: Tačnost, makro i mirko *F*-mera, respektivno. Osnovni model (*baseline*) koji sve instance svrstava u najzastupljeniju klasu skupa podataka ima tačnost 74.0%, makro *F*-meru 42.7% i mikro *F*-meru 74.6%. Broj anotiranih primera korišćenih u *All* postavci je 541, a broj anotiranih primera korišćenih u *L* postavci i ko-trening postavkama je 6.



Grafikon 25 *Korisnik*$_2$: Tačnost, makro i mirko *F*-mera, respektivno. Osnovni model (*baseline*) koji sve instance svrstava u najzastupljeniju klasu skupa podataka ima tačnost 74.6%, makro *F*-meru 42.7% i mikro *F*-meru 74.4%. Broj anotiranih primera korišćenih u *All* postavci je 576, a broj anotiranih primera korišćenih u *L* postavci i ko-trening postavkama je 6.



Grafikon 26 *Korisnik*$_3$: Tačnost, makro i mirko *F*-mera, respektivno. Osnovni model (*baseline*) koji sve instance svrstava u najzastupljeniju klasu skupa podataka ima tačnost 67.4%, makro *F*-meru 40.2% i mikro *F*-meru 67.2%. Broj anotiranih primera korišćenih u *All* postavci je 256, a broj anotiranih primera korišćenih u *L* postavci i ko-trening postavkama je 6.



Grafikon 27 *Korisnik*$_4$: Tačnost, makro i mirko *F*-mera, respektivno. Osnovni model (*baseline*) koji sve instance svrstava u najzastupljeniju klasu skupa podataka ima tačnost 75.9%, makro *F*-meru 43.1% i mikro *F*-meru 75.9%. Broj anotiranih primera korišćenih u *All* postavci je 249, a broj anotiranih primera korišćenih u *L* postavci i ko-trening postavkama je 6.

Grafikoni 24 – 27 i tabela 20 pokazuju da je ponašanje poređenih algoritama slično za sve korisnike. Možemo izvući sledeće zaključke:

- Od svih postavki koje koriste samo jedan od podskupova obeležja (kolone *Korisnik*, *Žanr* i *Radnja* u tabeli 20), postavka bazirana na obeležjima *korisničkog* pogleda je postigla najveću tačnost i *F*-meru. Ovo nije iznenađujuće budući da algoritmi bazirani na kolaborativnom filtriranju generalno postižu bolje performanse od algoritama baziranih na filtriranju

sadržaja. Takođe, u ovde predstavljenim eksperimentima su za predstavljanje sadržaja filma korišćena veoma bazična obeležja koje je moguće pobojšati.

- Najslabija od postavki baziranih na jednom pogledu je postavka bazirana na *pogledu radnje*. Obeležja ovog pogleda su bazirana na opisima zapleta skinutih sa IMDB web sajta i, kao što je primećeno u [Qu 2013], ovi opisi su generalno kratki, što rezultuje retkim skupom podataka i verovatno predstavlja uzrok loših performansi klasifikatora treniranog na *pogledu radnje*.
- Za postavke u kojima se kombinacije različitih podskupova obeležja tretiraju kao jedinstven skup obeležja (*L* i *All*), kombinovanje *korisničkog* pogleda sa drugim pogledima (tj. pogledi *korisnik_radnja, korisnik_žanr* i *korisnik_žanr_radnja*) veoma malo pobojšava preformanse u odnosu na postavku gde se koriste isključivo obeležja korisničkog pogleda.
- Za postavke *L* i *All* kombinacija pogleda *žanra* i *radnje* (tj. pogled *žanr_radnja*) ima najslabije performanse, u nekim slučajima čak slabije od performansi koje se dobijaju korišćenjem isključivo jednog od pogleda *žanr* ili *radnja*. Čak i u slučaju da raspolažemo sa klasnim obeležjima celog skupa podataka (kombinacija *žanr_radnja* u *All* postavci), postižu se slabije performanse u odnosu na postavku baziranu na obeležjima korisničkog pogleda gde se koristi mali skup obeležja (*L* postavka u kojoj se koristi *korisnički pogled*). Ovo nije iznenađujuće jer je u [Qu 2013] istaknuto da postoji snažna korelacija između žanra filma i reči koji se nalaze u opisu njegove radnje, što može biti uzrok slabih performansi.

Međutim, iako obeležja iz pogleda *žanra* i *radnje* deluju beskorisna u poređenju sa *korisničkim* pogledom kada se koriste na taj način što se sva obeležja kombinuju u jedinstven pogled, iz rezultata sledi da su ovi pogledi veoma korisni u ko-trening postavkama koje koriste dva odvojena pogleda:

- Za sve kombinacije pogleda, *Natural* postavka je uspela da unapredi performanse u odnosu na slabi polazni klasifikator (*L*), odnosno performanse koje se postižu primenom *Natural* postavke su bolje od performansi *L* postavke i u slučaju kada se za *L* postavku koristi data ista kombinacija pogleda kao za *Natural* i u slučaju kada se za *L* postavku koriste pojedniačni pogledi (npr. *Natural* primenjen na kombinaciju pogleda *korisnik_žanr* postiže bolje performanse of *L* postavke primenjene na kombinaciju *korisnik_žanr*, al i od *L* postavke primenjene na poglede *korisnik* i *žanr* ponaosob). Takođe, za sve kombinacije pogleda osim *žanr_radnja* kombinacije performanse *Natural* postavke su u rangu performansi *All* postavke. Ovo znači da smo obukom ko-treninga na veoma malom broju anotiranih primera postigli performanse koje bi smo bili u stanju da dostignemo kada bi smo raspolagali velikim brojem anotiranih primera. Za *žanr_radnja* kombinaciju pogleda performanse *Natural* postavke su nešto lošije od performansi *All* postavke za istu kombinaciju pogleda, ali je i dalje postignuto značajno unapređenje performansi u odnosu na početni klasifikator (*L* postavka).

- Kao što je i očekivano, *Random* postavka postiže lošije performanse u odnosu na *Natural* postavku. *MV* postiže bolje performanse od *Random* i *Natural* i, u nekim slučajevima, čak prelazi performanse *All* postavke u smislu postignute tačnosti, mikro i makro *F*-mere. *RSSalg* postavka je postigla nesto slabije performanse od *MV* postavke, ali je takođe u rangu *All* postavke. Konačno, *RSS_best* postavka postiže najbolje performanse od svih poređenih postavki.
- Gledano po kombinacijama pogleda, najbolje peroformanse su postignute kombinacijama *korisnik_žanr* i *korisnik_žanr_radnja*.
- Kombinacija *žanr_radnja* daje najgore performanse u poređenju sa ostalim postavkama koje koriste dva pogleda. *RSS* i *RSS_best* postavke su korišćenjem kombinacije pogleda *žanr_radnja* postigle unapređenje performansi u odnosu na *All* postavku za istu kombinaciju obeležja *žanr_radnja*. Ovo je konzistentno sa rezultatima prikazanim u odeljku 4.4 – *RSS* postiže najbolje performanse u slučaju da postoji velika redudantnost u skupu podataka, što jeste slučaj za *žanr_radnja* kombinaciju obeležja gde postoji korelacija između različitih obeležja.
- U slučaju kombinacije *žanr_radnja* performanse *RSS_best* postavke su su u rangu performansi *All* postavke primenjene na *žanr* pogled (koja je opet bolja od *All* postavke primenjene na kombinaciju pogleda *žanr_radnja*). Međutim, u nekim slučajevima (npr. $Korisnik_2$), ove performanse su slabije od performansi koje *L* postavka postiže korišćenjem *korisničkog* pogleda. Međutim, potrebno je istaći da ova kombinacija pogleda i dalje može biti veoma korisna. Uzmimo u obzir situaciju gde je novi korisnik ocenio samo filmove koje ni jedan drugi korisnik nije ocenio. U ovoj situaciji, primena ko-treninga ili, bolje *RSSalg* algoritma, sa ovom kombinacijom pogleda može značajno da unapredi performanse inicijalnog klasifikatora. Konačno, neophodno je napomenuti da su ovde izloženi samo inicijalni eksperimenti. U budućnosti je potrebno unaprediti obeležja koja se koriste u klasifikatoru baziranom na filtriranju sadržaja, npr. uključivanjem semantike [Qu 2013].

6.3.4 Zaključak

U ovom odeljku je izloženo kako se primenom ko-treninga može pristupiti problemu pojave novog korisnika u sistemu za davanje preporuka, odnosno situaciji u kojoj je zbog nedostatka istorije ocenjivanja datog korisnika teško dati kvalitetne personalizovane predikcije.

Problem preporuke je postavljen kao problem klasifikacije. Za svakog korisnika, gradi se poseban klasifikacioni model kojim je moguće predvideti da li će se artikl koji korisnik nije ocenio svideti datom korisnku (te ga treba preporučiti) ili neće. U cilju olakšavanja problema pojave novog korisnika, dizajniran je hibridni sistem za preporuke koji kao jedan pogled na skup podataka koristi ocene drugih korisnika, a kao drugi pogled koristi obeležja bazirana na opisu artikla.

U ovde izloženim eksperimentima dati algoritam je primenjen na problem preporuke filmova i korišćen je popularan *MovieLens* skup podataka. Za opis artikla korišćena je informacija o žanrovima kojima film pripada, kao i opis radnje filma. Testirano je nekoliko različitih ko-trening postavki koje su primenjene na različite kombinacije pogleda. U ovde izvedenim eksperimentima je primenom ko-treninga postignuto unapređenje performansi u odnosu na inicijalni klasifikator (model treniran primenom algoritma nadgledanog obučavanja isključivo na anotiranim podacima), a takođe su dostignute performanse koje bi smo mogli postići nadgledanim običavanjem u slučaju da raspolažemo anotacijama za sve trening podatke (i anotirane i neanotirane koji se takođe koriste u polu-nadgledanoj postavci). Najbolje performanse u eskperimentima su postignute korišćenjem originalnog ko-trening algoritma i *RSSalg* algoritma koji su primenjeni korišćenjem obeležja baziranih na ocenama drugih korisnika kao prvog pogleda i kombinacije osobina izvedenih iz žanra i opisa radnje filma kao drugog pogleda, ili korišćenjem obeležja baziranih na ocenama drugih korisnika kao prvog pogleda i osobina izvedenih iz žanra kao drugog pogleda. Eksperimenti su takođe pokazali da ko-trening prediktor baziran na filtriranju sadržaja i konstruisan isključivo na osnovu obeležja izvedenih iz žanra i radnje filma može biti veoma koristan u slučaju da ne raspolažemo ocenama drugih korisnika.

Najvažniji zaključak izveden iz eksperimenata je da se primenom ko-treninga, polazeći od svega 6 ocenjenih filmova mogu dostići performanse koje bi dostigao nadgledani sistem ukoliko bi raspolagao velikom istorijom rejtinga za datog korisnika.

Ovde prikazani rezultati su samo preliminarni eksperimenti izvedeni sa ciljem da se dobije generalna ideja može li se predloženi hibridni sistem koristiti za olakšavanje problema pojave novog korisnika u sistemu za davanje preporuka. Postoji mnogo načina na koje se ovaj pristup može pobojšati. Prvo, model baziran na filtriranju sadržaja, korišćen kao drugi pogled u prikazanoj postavci, bi se mogao obogatiti ekstrakcijom semantički bogatijih obeležja kao što je predloženo u [Qu 2013]. Takođe, u ovde izvedenim eksperimentima za pogled baziran na ocenama drugih korisnika je korišćen samo podskup od 50 korisnika koji su dali najviše ocena u celom sistemu. U budućnosti bi se trebalo eksperimentisati sa mogućnostima primene raspoloživih ocena svih korisnika u sistemu. Konačno, potrebno je eksperimentisati i sa mogućnošću primene razvijenog sistema na problem pojave novih, neocenjenih, artikala u sistemu.

7 Zaključak

Predmet istraživanja ove disertacije je bio razvoj sistema za automatsku klasifikaciju podataka. Cilj kome se težilo je da sistem bude primenljiv u širokom spektru domena gde je neophodna klasifikacija podataka, pri čemu je teško, ili čak nemoguće, doći do dovoljno velikog i raznovrsnog obučavajućeg skupa podataka. U ovoj disertaciji su predstavljena dva modela koja rešavaju zadatak automatske klasifikacije u slučaju nepostojanja dovoljno velikog anotiranog korpusa za obuku. Oba modela su bazirana na ko-trening algoritmu koji predstavlja moćnu paradigmu za polu-nadgledano obučavanje, ali čija je široka primena ograničena zahtevom postojanja prirodne podele obeležja na dva odvojena skupa (pogleda). Zbog toga su modeli predstavljeni u disertaciji dizajnirani sa ciljevima omogućavanja primene ko-trening algoritma na skupove podataka bez prirodne podele obeležja i unapređenja njegovih performansi.

U prvom poglavlju ove disertacije je definisan zadatak automatske klasifikacije i obrazložena je potreba za sistemima opisanim u disertaciji. U ovom poglavlju su predstavljeni i osnovni modeli i metodologije za obuku automatskih klasifikatora i optimizaciju njihovih parametara koji su korišćeni u okviru ove disertacije. Takođe, ovo poglavlje daje i generalan pregled tehnika polu-nadgledanih obučavanja kojima predložena rešenja pripadaju.

Drugo poglavlje disertacije daje pregled dostupne literature koja se bavi rešavanjem problema primene ko-treninga u situacijama gde je adekvatna podela obeležja na dva pogleda nedostupna. U ko-treningu se dva pogleda koriste za obučavanje dva različita klasifikatora na istom polaznom skupu podataka. Nakon toga se, iterativno, svaki od klasifikatora primenjuje na skup neanotiranih zapisa sa ciljem anotacije zapisa za koje je anotacija najpouzdanija. Anotirani zapisi se dodaju u inicijalni obučavajući skup, nakon čega se oba klasifikatora ponovo obučavaju na uvećanom polaznom skupu podataka. U kombinaciji sa adekvatnom podelom obeležja ovaj pristup je veoma efektivan. Međutim, ovakva podela obeležja nije uvek definisana, a korišćenje neadekvatne podele obeležja može dovesti i do degradacije performansi u odnosu na polazni klasifikator.

U trećem poglavlju ove disertacije izložen je detaljan opis metodologije rešenja predloženih u ovoj disertaciji. Modeli predstavljeni u ovoj tezi pristupaju problemu nedostatka adekvatne podele obeležja time što formiraju grupu nezavisno obučenih ko-trening klasifikatora i kombinuju njihove predikcije u cilju dobijanja konačnog modela uvećanih performansi. U cilju formiranja skupa nezavisno obučenih ko-trening klasifikatora prvo se generiše predefinisani broj slučajnih podela obeležja polaznog skupa podataka. Svaka od generisanih podela se koristi kako bi se primenom ko-tening algoritma obučio jedan od klasifikatora grupe. Dobijena grupa klasifikatora se eksploatiše u cilju formiranja modela značajno uvečanih performansi u odnosu na polazni model.

Dva modela predstavljena u ovoj tezi, nazvana *Algoritam Statistike Slučajnih Podela* (***R**andom **S**plit **S**tatistics **Alg**orithm, RSSalg*) i *Integracija Višestrukih Ko-treninranih Klasifikatora* (**I**ntegration of **M**ultiple **C**o-trained **C**lassifiers, *IMCC*) se međusobno razlikuju po načinu kombinovanja predikcija grupe klasifikatora. Integracija predikcija se kod *RSSalg* modela vrši formiranjem uvećanog obučavajućeg skupa primenom popularne metode većinskog glasanja i naknadnim filtriranjem ovog skupa u cilju eliminacije pogrešno anotiranih instanci. Na ovako formiranom obučavajućem skupu se zatim nadgledanim procesom obučava finalni klasifikator. Kod *IMCC* modela se u cilju integracije predikcija svaki od obučenih ko-trening klasifikatora primenjuje na skup test instanci, nakon čega se koristi sofisticirana tehnika za nenadgledanu estimaciju tačnih klasnih obeležja u slučaju postojanja višestrukih anotatora nepoznatog kvaliteta.

Četvrto poglavlje ove disertacije izlaže metodologiju evaluacije predloženih modela i postignute rezultate. U cilju provere robustnosti predstavljenih modela, oni su testirani primenom metode unakrsne evaluacije na skupovima podataka različitih veličina, dimenzionalnosti i redudantnosti. U ove svrhe odabrana su 4 skupa podataka prirodnog jezika, kao i 13 *UCI* skupova podataka. Performanse modela su poređene sa performansama alternativnih ko-trening baziranih tehnika, dizajniranih sa istim ciljem primene ko-treninga na skupove podataka bez prirodne podele obeležja. Takođe, izvršeno je poređenje sa klasičnim ko-trening algoritmom primenjenim sa prirodnom podelom obeležja (u slučajevima gde je data podela poznata), kao i sa Naivnim Bajesovim modelom obučenom na malom inicijalnom anotiranom skupu i sa istim modelom obučenim na značajno većem anotiranom skupu. Izvršeni su i statistički testovi kako bi se utvrdilo da su dobijene razlike u tačnosti metoda značajne.

U izvedenim eksperimentima *IMCC* postavka je pobedila sve testirane alternative. Na svim testiranim skupovima podataka uspela je da unapredi performanse polaznog klasifikatora treniranog nadgledanim obučavanjem na malom anotiranom skupu podataka, a takođe je dostigla performanse klasifikatora treniranog nadgledanim obučavanjem na značajno većem obučavajućem skupu. Pokazano je da bi *RSSalg* postavka uz pažljivu optimizaciju parametara mogla dostići performanse *IMCC* postavke. Međutim, predloženi način automatske optimizacije parametara *RSSalg* metode se pokazao efektivnim samo na redudantnijim skupovima podataka. Bolji način optimizacije uvedenih parametara koji bi bio efektivan na većem broju domena ostaje zadatak za budućnost. Preostale testirane ko-trening alternative su se pokazale nepouzdane, u smislu da na većini skupova podataka nisu uspele da unaprede performanse polaznog klasifikatora.

U četvrtom poglavlju analiziran je i uticaj odabira vrednosti parametara na performanse predstavljenih modela. Pokazalo se da su predstavljeni modeli

prilično robustni na korišćeni broj slučajnih podela. Inicijalno, sa porastom broja podela (ko-trening klasifikatora u grupi) rastu i performanse datih postavki, a nakon određene tačke performanse modela stagniraju, odnosno, ne pobojšavaju se, niti se degradiraju dodavanjem novih ko-trening klasifikatora. Takođe, analizirani su uvedeni parametri koji utiču na eliminaciju pogrešno anotiranih instanci iz integrisanog skupa za obuku finalnog klasifikatora u *RSSalg* postavci. Utvrđeno je da svi ovakvi parametri imaju velik uticaj na performanse *RSSalg* postavke (da ih vredi zadržati). Pogrešan odabir vrednosti parametara može uzrokovati da *RSSalg* postavka nije u stanju da unapredi ili da čak degradira performanse polaznog klasifikatora. Optimalne vrednosti parametara su veoma osetljive na konkretne podatke, te se zbog kompleksnosti problema preporučuje se upotreba GA u njihovoj optimizaciji.

U cilju karakterizacije grupe skupova podataka na kojima je moguće pouzdano primeniti predstavljene modele meren je i uticaj redudantnosti dobijenih podela na performanse rešenja. Rezultati su ukazali da je *RSSalg* postavka prilično osetljiva na nedostatak redudantnosti pogleda nastalih slučajnim podelama obeležja, dok je *IMCC* postavka donekle robustnija.

Analiziran je i uticaj parametara ko-treninga na *RSSalg* postavku. U poređenju sa ko-trening alternativama, *RSSalg* postavka sa idealno odabranim parametrima je pokazala najmanju osetljivost na odabrane vrednosti parametara.

U petom poglavlju disertacije je opisana softverska arhitektura sistema koji implementira predstavljene modele i daje podršku za opisanu eksperimentalnu proceduru.

U šestom poglavlju je pokazano kako se predložena rešenja mogu primeniti na nekoliko realnih problema gde nedostatak anotiranih korpusa predstavlja veliki problem: detekciju subjektivnosti teksta, višekategorijsku klasifikaciju i u okviru sistema za davanje preporuka.

Nedostatak predloženih *IMCC* i *RSSalg* modela jeste vremenska kompleksnost. Međutim, ovi algoritmi nisu predviđeni da rade u realnom vremenu u interakciji sa korisnikom (rade u potpunosti *offline*), što čini njihovu vremensku kompleksnost manje problematičnom. Takođe, sa obzirom da je obučavanje pojedinačnih ko-trening klasifikatora u potpunosti nezavisno, rešenje je moguće u značajnoj meri paralelizovati.

Pokazano je da postavke predložene u ovoj disertaciji postižu dobre performanse, naročito na skupovima podataka koji se odlikuju velikom redudantnošću obeležja. Ipak, uočeno je i dosta mogućnosti za pobojšanje opisanog pristupa koje će biti tema budućeg rada.

U predloženim metodologijama je procedura generisanja slučajnih podela obeležja u potpunosti slučajna. U budućnosti bi se performanse rešenja potencijalno mogle značajno unaprediti razvojem heurističkog pristupa za izbor primenjenih podela obeležja. Kao što je diskutovano u odeljku 4.5.3, potrebno je

istražiti mogućnosti pronalaženja skupa podela koje bi rezultovale ko-trening klasifikatorima koji su što tačniji, a pri tome međusobno što više različiti kako bi mogli da uče jedni od drugih. Potencijalno, na ovaj način bi se mogla vršiti i automatska adaptacija broja korišćenih podela u zavisnosti od podataka.

Prilikom integracije višestrukih anotiranih skupova u *RSSalg* metodologiji se koristi jednostavno većinsko glasanje. Ova metoda bi se mogla zameniti boljim pristupom. Na primer, mogla bi biti korišćena *GMM-MAPML* metoda kao u *IMCC* pristupu. Međutim, ova metoda pretpostavlja da nemamo nikakvo znanje o anotatorima, dok bi se u slučaju ko-trening klasifikatora mogla koristiti i njihova pouzdanost u datu anotaciju, tako i pouzdanost u sam ko-trening klasifikator, što bi se moglo izraziti preko mera predloženih u [Opitz 1999] i [Salalhedin 2010].

Jedan od nedostataka predstavljenog *RSSalg* algoritma jeste njegova velika osetljivost na pragove koji služe za filtriranje pogrešno anotiranih instanci. Predložena metodologija automatske detekcije pragova se pokazala uspešna na redudantnijim skupovima podataka, međutim omanula je na manje redudantnim UCI skupovima. Jedna mogućnost pobojšanja je korišćenje unakrsne validacije za procenu tačnosti rezultujućeg modela, nasuprot korišćenju jednog test skupa sastavljenog od potencijalno pogrešno anotiranih instanci. Međutim, ovo je prilično vremenski skupa procedura. Drugi predlog je da se istraži mogućnost povezivanja datog rešenja sa mehanizmima editovanja podataka (*data editing mehanisms*) u cilju identifikacije pogrešno anotiranih instanci. Na primer, jedan ovakav sistem je opisan u [Muhlenbach 2004].

Uočeno je i da *RSSalg* postavka, čak i sa idealno odabranim parametrima, na određenim manje redudantnim skupovima podataka ne može da dostigne performanse klasifikatora nastalog obučavanjem na svim anotiranim podacima. Za ovakve skupove podataka je potrebno istražiti mogućnost dodavanja novih obeležja u cilju povćanja redudancije, kao što je predloženo u [Breiman 2001].

Konačno, zadatak za budućnost je i istraživanje mogućnosti primene predloženih rešenja na još neke realne probleme gde nedostatak anotiranih korpusa u značajnoj meri smanjuje mogućnosti razvoja sistema automatske klasifikacije. Na primer, automatska ekstrakcija metapodataka iz naučnih publikacija bi bila od velike koristi u okviru informacionog sistema za nadgledanje aktivnosti naučnog istraživanja, kao što je CRIS UNS sistem na Univerzitetu u Novom Sadu [Kovačević 2011a]. Takođe, automatska detekcija i sumarizacija metodologije naučnih radova bi u mnogome pomogla istraživačima prilikom pregleda postojeće literature [Kovačević 2011b]. Predloženi pristup bi se potencijalno mogao kombinovati i sa aktivnim učenjem. Veoma jednostavan način bi bio da se instance koje su usled nepouzdanosti anotacije eliminisane iz obučavajučeg skupa u *RSSalg* postavci proslede anotatoru, budući da bi upravo

ovakve instance za koje postoji veliko neslaganje klasifikatora mogle biti naročito informativne.

8 Literatura

[Adomavicius 2005] Adomavicius, G., 2005. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the- Art and Possible Extensions. In *IEEE Transactions on Knowledge and Data Engineering*, vol 17, no.6.

[Back 1991] Back and F. Hoffmeister. Extended Selection Mechanisms in Genetic Algorithm. In: Belew, R. K. & Brooker, L. B., editors, *Proc. 4th International Conference on Genetic Algorithm*, pp. 92-99, 1991.

[Billsus 1998] D. Billsus, and M. Pazzani. Learning collaborative information filters. *In Proceedings of Internationall Conference on Machine Learning*, Morgan Kaufmann Publishers, 1998.

[Bishop 2006] C. Bishop, C.: Pattern recognition and machine learning, pp. 203–213. Springer, New York, 2006.

[Blum 1998] A. Blum, and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning theory COLT'98*, ACM, pp 92-100, 1998.

[Blum 2001] A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the 18th International Conference on Machine Learning*, Williamston, MA, pp.19–26, 2001.

[Breiman 1996] L. Breiman. Bagging predictors. *Machine Learning*, vol. 24(2), pp. 123-140, 1996.

[Breiman 1996b] L. Breiman: Out-of-bag estimation, Technical Report, Statistics Department, University of California, 1996

[Breiman 2001] L. Breiman. Random forests. *Machine Learning*, 45(1), pp. 5-32, 2001.

[Castelli 1995] V. Castelli and T. Cover. The exponential value of labeled samples. *PatternRecognition Letters*, *16*, 105–111, 1995.

[Castelli 1996] V. Castelli and T. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory*, *42*, 2101–2117, 1996.

[Chan 2004] J. Chan, I. Koprinska and J. Poon. Co–training with a single natural feature set applied to email classification. *In Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence (WI'04)*, pages 586–589, Beijing, China 2004.

[Chan 2004b] J. Chan, I. Koprinska and J. Poon. Co-training on Textual Documents with a Single Natural Feature Set. In P. Bruza, A. Moffat and A. Turpin, eds., *Proceedings of the 9th Australasian Document Computing Symposium (ADCS'04)*, Department of Computer Science and Software Engineering, University of Melbourne, Melbourne, Australia, pp. 47-54, 2004.

[Chen 2010] Y. Chen, M. Harper, J. Konstan, and X. Li. Social Comparisons and Contributions to Online Communities: A Field Experiment on MovieLens. *American Economic Review* 100(4), 2010.

[Chapelle 2006] O. Chapelle, B. Schoelkopf and A. Zien (eds). Semi-Supervised Learning, MIT Press, 2006.

[Collins 1999] M. Collins and Y. Singer. Unsupervised models for named entity classifications. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, College Park, MD, pp.100–110, 1999.

[Cover 1965] T. Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. In *IEEE Transactions On Electronic Computers*, pp. 326-334, 1965.

[Cozman 2003] F. Cozman, I. Cohen and M. Cirelo. Semi-supervised learning of mixture models. *ICML-03, 20th International Conference on Machine Learning*, 2003.

[Dara 2002] R. Dara, S. Kremer and D. Stacey. Clustering unlabeled data with SOMs improves classification of labeled real-world data. *Proceedings of the World Congress on Computational Intelligence (WCCI)*, 2002.

[Delgado 1999] J. Delgado and N. Ishii, „Formal Models for Learning of User Preferences, a Preliminary Report," *Proc. Int'l Joint Conf. On Artificial Intelligence (IJCAI-99)*, Stockholm, Sweden, July, 1999.

[Dempster 1977] A. Dempster, N. Laird and Rubin, D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*. 1977.

[Demšar 2006] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, Vol. 7, pp. 1-30, 2006.

[Demriz 1999] A. Demiriz, K. Bennett and M. Embrechts. Semi-supervised clustering using genetic algorithms. *Proceedings of Artificial Neural Networks in Engineering*, 1999.

[Dietterich 1991] T.G. Dietterich and G. Bakiri. Error-Correcting output codes: a general method for improving multi-class inductive learning programs. In *Dean TL, McKeown K, eds. Proceedings of the 9th AAAI National Conference on Artificial Intelligence*, AAAI Press, CA, pp. 572-577, 1991.

[Dietterich 1995] T. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. Journal of Artificial Intelligence Research, vol. 2, pp. 263-286, 1995.

[Dietterich 1998] T.G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, Vol. 10, pp. 1895-1924, 1998.

[Dietterich 2000] T. G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine Learning, 40(2), pp. 139-157, 2000.

[Du 2010] J. Du, C. Ling and Z. Zhou. When Does Co-Training Work in Real Data?. *IEEE Trans. On Knowledge and Data Engineering*, 23(5), pp. 788-799, 2010. ISSN 1041-4347, 2010.

[Enright 2002] Enright, A.J., Van Dongen ,S. And Ouzounis,C.A., 2002. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res* , 30(7):1575-1584

[Feger 2006] F. Feger and I. Koprinska. Co–training Using RBF Nets and Different Feature Splits. In *Proceedings of 2006 International Joint Conference on Neural Network*, pp. 1878–1885, 2006.

[Fisher 1959] R.A. Fisher. Statistical methods and scientific inference. 2nd edn. Hafner Publishing Co, New York, 1959.

[Fraley 1998] C. Fraley and A.E. Raftery. How many clusters? Which clustering method? Answers via model-based cluster analysis. Computing Journal, pp. 578–588, 1998.

[Freund 1996] Y. Freund and R.E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of 13th Int. Conf. on Machine Learning (ICML'97)*, San Francisco: Morgan Kaufmann, pp. 148-156, 1996.

[Freund 1997] Y. Freund, H. Seung, E. Shamir, N. Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2-3), pp. 133-168, 1997.

[Friedman 1937] M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association, Vol. 32, pp. 675–701, 1937.

[Friedman 1940] M. Friedman. A comparison of alternative tests of significance for the problem of m rankings," Ann Math Stat 11:86–92, 1940.

[Garcia-Pedrajas 2008] N. Garcia-Pedrajas and D. Ortiz-Boyer. Boosting Random Subspace Method. *Neural Networks*, 21, pp. 1344–1362, 2008.

[Garcia 2008] S. Garcia and F. Herrera. An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. *Journal of Machine Learning Research*, 9:2677–2694, 2008

[Ghani 2002a] R. Ghani, and A. Fano. Building recommender systems using a knowledge base of product semantics. In *Proceedings of Workshop on Recommendation and Personalization in E-Commerce, at the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*, Malaga, Spain, May 2002.

[Ghani 2002b] R. Ghani. Combining Labeled and Unlabeled Data for Multiclass Text Categorization. In *Proceedings of the 19th International Conference on Machine Learning,* 2002.

[Goldberg 1989] D. Goldberg. *Genetic algorithms in search, optimization and machine learning*. Addison–Wesley, 1989.

[Goldberg 2009] A. Goldberg and X. Zhu. Keepin' it real: "Semi-supervised learning with realistic tuning", *Proceedings NAACL HLT Workshop on Semi-supervised Learning for Natural Language Processing*, pp. 19–27, 2009.

[Goldman 2000] S. Goldman and Y. Zhou. Enhancing supervised learning with unlabeled data. In *Proceedings of 17th International Conference on Machine Learning,* pp. 327–334, MorganKaufmann, San Francisco, CA, 2000.

[Guo 2012] Y. Guo and D. Schuurmans. Semi-supervised Multi-label Classification. Book section *Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science*, Vol. 7524, pp 355-370, 2012.

[Hall 2009] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I.H. Witten. The weka data mining software: an update. SIGKDD Explorations, 11, 10–18, 2009.

[Hady 2008a] M.F.A. Hady and F. Schwenker. Co-training by Committee: A New Semi-supervised Learning Framework. In *Proceedings of IEEE International Conference on Data Mining Workshops*, pp. 563-572, 2008.

[Hady 2008b] M.F.A. Hady, F. Schwenker and G. Palm. Semi-Supervised learning of tree-structured rbf networks using co-training. *In Proceedings of the 18th International Conference on Artificial Neural Networks (ICANN'08)*, Prague, Czech Republic, LNCS 5163, Springer-Verlag, pp. 79-88, 2008.

[He 2010] Y. He*.* Bayesian Models for Sentence-Level Subjectivity Detection*. Technical report kmi-10-02, Knowledge Media Institute, The Open University,* 2010*.*

[Ho 1998] T. Ho. The random subspace method for constructing decision forests. *IEEE Transactions Pattern Analysis and Machine Intelligence*, 20(8), pp. 832-844, 1998.

[Huang 2010] J. Huang, J.S. Shirabad, S. Matwin and J. Su. Improving Co-training with Agreement-Based Sampling. In *Proceeding RSCTC'10 Proceedings of the 7th international conference on Rough sets and current trends in computing*, pp 197-206, 2010.

[Holland 1975] J. H. Holland. Adaptation in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor, 1975.

[Hwa 2003] R. Hwa, M. Osborne, A. Sarkar, and M. Steedman. Corrected cotraining for statistical parsers," In *Working Notes of the ICML'03 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, Washington, DC, 2003.

[Jannach 2013] D. Jannach, L. Lerche, F. Gedikli, and G. Bonnin, „What recommenders recommend – An analysis of accuracy, popularity, and sales diversity effects," *21st Int'l Conf. User Modeling, Adaptation and Personalization (UMAP 2013),* Rome, Italy, 2013.

[Joachims 1999] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conf. on Machine Learning*, pp. 200–209. Morgan Kaufmann, San Francisco, CA, 1999.

[Joachims 2001] T. Joachims. A Statistical Learning Model of Text Classification for Support Vector Machines. In *Proceedings of the 24th ACM–SIGIR international conference on research and development in information retrieval*, pages 128–136, 2001

[Karger 1996] D. R. Karger and C. Stein. A New Approach to the Minimum Cut Problem. Journal of the ACM, 43, pp. 601-640, 1996.

[Kernigham 1970] B. W. Kernigham and S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs. The Bell System Technical Journal, 49, pp. 291-307, 1970.

[Kiritchenko 2001] S. Kiritchenko and S. Matwin. Email classification with co-training. *In Proceedings of the 2001 Conference of the Centre for Advanced Studies on Collaborative Research (CASCON'01)*, IBM Press, pp. 8-19, 2001.

[Koprinska 2007] I. Koprinska, J. Poon, J. Clark and J. Chan. Learning to Classify E–mail. *Information Sciences*, 177:2167–2187, Elsevier, 2007.

[Kou 2004] Kou, Y. , Lu, C. –T. , Sirwongwattana, S. , Huang, Y. –P, 2004. Survey of fraud detection techniques. In *Proceedings of the IEEE International Conference on Networking, Sensing and Control*.

[Kovačević 2011a] A. Kovačević, D. Ivanović, B. Milosavljević, Z. Konjović and D. Surla. Automatic extraction of metadata from scientific publications for CRIS systems. Program: Electronic library and information systems, 45(4), 376-396, 2011.

[Kovačević 2011b] A. Kovačević, Z. Konjović, B. Milosavljević and G. Nenadic. Mining methodologies from NLP publications: A case study in automatic terminology recognition. Computer Speech & Language, 26(2), 105-126, 2011.

[Kuncheva 2003] L. Kuncheva, C.J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2): pp. 181-207, 2003.

[Lawrence 2005] N.D. Lawrence and M.I. Jordan. Semi-supervised learning via Gaussian processes. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press, 2005.

[Levin 2003] A. Levin, P. Viola and Y. Freund. Unsupervised improvement of visual detectors using co-training. In *Proceedings of the International Conference on Computer Vision*, 2003.

[Lewis 1993] D. Lewis and M. Ringuette. A comparison od two learning algorithms for text categorization. In: *Third Annual Symposium on Document Analysis and Information Retrivial*, pp 392-401, 1993.

[Li 2007] M. Li and Z.H. Zhou. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Trans. On Systems, Man and Cybernetics*, 37(6), pp. 1088-1098, 2007.

[Linden 2003] G. Linden, B. Smith, and J. York. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. In *IEEE Internet Computing* 7(1), 76–80, 2003.

[MacKay 2003] D. MacKay. Dependent Random Variables. *Information Theory, Inference and Learning Algorithms* Cambridge Univ. Press, 2003.

[Maeireizo 2004] B. Maeireizo, D. Litman and R. Hwa. Co-training for predicting emotions with spoken dialogue data. *The Companion Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2004.

[Mao 2006] Y. Mao and G. Lebanon. Isotonic conditional random fields and local sentiment flow. In *Neural Information Processing Systems* (NIPS), 2006.

[Martinez 2004] W.L. Martinez and A.R. Martinez. Exploratory data analysis with MATLAB, pp. 163–195. Chapman & Hall/CRC, Boca Raton, 2004.

[Melville 2003] P. Melville and R.J. Mooney. Constructing diverse classifier ensembles using artificial training examples. In *Proceedings of the 18th international joint conference on Artificial intelligence* (IJCAI'03). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 505-510, 2003.

[McDonald 2007] R. McDonald, K. Hannan, T. Neylon, M. Wells and J. Reynar. Structured models for fine-to-coarse sentiment analysis. In *Annual Meeting of the Association for computational Linguistics* (ACL)*,* 2007.

[McCallum 1998] McCallum and K. Nigam, K. A comparison of event models for naivebayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*., 1998.

[Muhlenbach 2004] F. Muhlenbach, S. Lallich, and D.A. Zighed. Identifying and handling mislabelled instances. *Journal of Intelligent Information Systems*, vol. 22, no.1, pp.89–109, 2004.

[Mihalcea 2004] R. Mihalcea. Co-training and Self-training for Word Sense Disambiguation. In *Proceedings of the Conference on Natural Language Learning (CoNLL 2004)*, pp. 33-40, 2004.

[Munkhdalai 2012] T. Munkhdalai, M. Li, U. Yun, O. Namsrai, and K.H. Ryu. An Active Co-Training Algorithm for Biomedical Named-Entity Recognition. *Journal of Information Processing Systems*, Vol. 8, Issue 4, pp. 575-588, 2012.

[Murray 2010] G. Murray and G. Carenini. Subjectivity Detection in Spoken and Written Conversations. *Journal of Natural Language Engineering (JNLE),* 2010.

[Muslea 2002] I. Muslea, S. Minton and C. Knoblock. Active + Semi-Supervised Learning = Robust Multi-View Learning. *In Proceedings ICML*, 2002.

[Narayanan 2007] H. Narayanan, M. Belkin and Partha Niyogi. On the Relation Between Low Density Separation, Spectral Clustering and Graph Cuts. *Advances in Neural Information Processing Systems 19,* MIT Press, Cambridge, MA*,* 2007.

[Ng 2003] V. Ng and C. Cardie. Weakly supervised natural language learning without redundant views. *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 94–101, Edmonton, USA, 2003.

[Nigam 2000a] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell, 2000. Text classification from labeled and unlabeled documents using EM. In *Machine Learning,* Special issue on information retrieval, pp. 103–134.

[Nigam 2000b] K. Nigam and R. Ghani, 2000. Understanding the behavior of co–training. In *Proceedings of the Workshop on Text Mining at the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD–2000)*.

[Nigam 2001] K. Nigam. *Using unlabeled data to improve text classification* (Technical Report CMU-CS-01-126). Carnegie Mellon University. Doctoral Dissertation. 2001

[Moody 1989] J. Moody and C. Darken. Fast training in networks of locally-tunedprocessing units. Neural Computing, vol. 1, pp. 284-294, 1989.

[Munkhdalai 2012] T. Munkhdalai, M. Li, U. Yun, O. Namsrai, and K.H. Ryu. An Active Co-Training Algorithm for Biomedical Named-Entity Recognition. In *JIPS*, pp.575-588, 2012.

[Opitz 1999] D. Opitz. Feature selection for ensembles. In *Proceedings of the 16th International Conference on Artificial Intelligence*, pp. 379–384, 1999.

[Pang 2002] B. Pang, L. Lee and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of EMNLP, 2002.*

[Pang 2004] B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Annual Meeting of the Association for Computational Linguistics (ACL), 2004.*

[Pang 2008] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, 2008.

[Park 1991] J. Park and I. Sandberg. Universal approximation using radial-basisfunction networks. Neural Computation, vol. 3, pp. 246-257, 1991.

[Pierce 2001] D. Pierce and C. Cardie. Limitations of Co–Training for Natural Language Learning from Large Datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP–2001),* 2001.

[Porter 1980] M.F. Porter. An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137, 1980.

[Qu 2013] W. Qu, K-S. Song, Y-F. Zhang, S. Feng, D-L. Wang, and G. Yu. A Novel Approach Based on Multi-View Content Analysis and Semi-Supervised Enrichment for Movie Recommendation. Journal of Computer Science and Technology 28(5): 776-787, September 2013.

[Quinlan 1993] R.J. Quinlan. C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., 1993.

[Ratsaby 1995] J. Ratsaby and S. Venkatesh. Learning from a mixture of labeled and unlabeled examples with parametric side information. *Proceedings of the Eighth Annual Conference on Computational Learning Theory*, 412–417, 1995.

[Resnick 1997] P. Resnick, and H. Varian. Recommender Systems. In *Communications of the ACM 40(3)*, 56–58, 1997.

[Riloff 1999] E. Riloff and R. Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on Artificial Intelligence*, Orlando, FL, pp.474–479, 1999.

[Riloff 2003a] E. Riloff, J. Wiebe and T. Wilson. Learning subjective nouns using extraction pattern bootstrapping. *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, 2003.

[Riloff 2003b] E. Riloff and J. Wiebe. Learning extraction patterns for subjective expressions. In *EMNLP, 2003.*

[Rosset 2005] S. Rosset, J. Zhu, H. Zou and T. Hastie. A method for inferring label sampling mechanisms in semi-supervised learning. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press, 2005.

[Rosenberg 2005] C. Rosenberg, M. Hebert and H. Schneiderman. Semi-supervised self-training of object detection models. *Seventh IEEE Workshop on Applications of Computer Vision*, 2005.

[Salaheldin 2010] A. Salaheldin and N. El Gayar. New feature splitting criteria for co–training using genetic algorithm optimization. *Lecture Notes in Engineering and Computer Science*, 5997/2010: 22–32, 2010.

[Salton 1980] G. Salton and C. Buckley. Term–weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[Sarkar 2001] A. Sarkar. Applying co-training methods to statistical parsing. In *Proceedings of the 2nd Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, pp. 95-102, Pittsburgh, PA, 2001.

[Schuurmans 2001] D. Schuurmans and F. Southey. Metric-based methods for adaptive model selection and regularization. *Machine Learning, Special Issue on New Methodsfor Model Selection and Model Combination*, *48*, pp. 51–84, 2001.

[Seeger 2001] M. Seeger. *Learning with labeled and unlabeled data* (Technical Report). University of Edinburgh, 2001.

[Settles 2010] B. Settles. Active Learning Literature Survey. *Computer Sciences Technical Report 1648. University of Wisconsin–Madison*, retrieved 2010.

[Shazmeen 2013] S.F. Shazmeen, M.M.A. Baig and M.R. Pawar. Regression Analysis and Statistical Approach on Socio-Economic Data. International Journal of Advanced Computer Research, Vol. 3, No 3(11), 2013.

[Shinnou 2004] H. Shinnou and M. Sasaki M. Semi-supervised learning by Fuzzy clustering and Ensemble learning. In the *4th international conference on Language Resources and Evaluation (LREC2004)*, Academic Journal 2004., pp 399-402, 2004.

[Slivka 2010] J. Slivka, A. Kovačević and Z. Konjović. Co-training based algorithm for datasets without the natural feature split. *Proceedings of the 2010 8th International Symposium on Inteligent Systems and Informatics (SISY), p.p. 279-284, 2010.* ISBN: 9781424473953

[Slivka 2011a] J. Slivka, A. Kovačević and Z. Konjović. Multi-label 155lassification experiments with co-training based algorithm. In *Proceedings of the International Conference on Information Society Technology and Management (ICIST)*, 2011.

[Slivka 2011b] J. Slivka, A. Kovačević and Z. Konjović. "Multi-Label Classification Experiments with Co-Training Based-Algorithm", *E-society journal*, Vol. 2, No 1, pp. 77-87, 2011.

[Slivka 2012a] J. Slivka, A. Kovačević and Z. Konjović. Co-training based-algorithms applied to subjectivity detection task. In *Proceedings of the International Conference on Information Society Technology and Management (ICIST)*, 2012.

[Slivka 2012b] J. Slivka, Z. Ping, A. Kovačević, Z. Konjović an Z. Obradović. Semi-Supervised Learning on Single-View Datasets by Integration of Multiple Co-trained Classifiers. In *Proceedings of the* 11th *International Conference on Machine Learning and Applications* (*ICMLA*), Boca Raton: The institute of Electrical and Electronic Engineers, Inc., 12-15 December, pp. 458-464, 2012.

[Slivka 2013a] J. Slivka, A. Kovačević, and Z. Konjović. Combining co-training with ensemble learning for application on single-view natural language datasets. *Acta Polytechnica Hungarica*, Vol. 10, No 2, pp. 133-152, 2013.

[Slivka 2013b] J. Slivka, A. Kovačević. Semi-Supervised News Genre Classification. *The IPSI BgD Transactions on Internet Research*, Vol. 9, No 1, pp. 32-37, ISSN 1820-4503, 2013.

[Slivka 2014] J. Slivka, A. Kovačević and Z. Konjović. Addressing the cold-start new-user Problem for Recommendation with Co-traning. *In Proceedings of International Conference Internet Society Technology and Management*, Serbia, 2014.

[Sokolova 2009] M. Sokolova, and G. Lapalme. A systematic analysis of performance measures for classification tasks. Inf. Process. Manage, 45(4): 427-437, 2009.

[Steedman 2003] M. Steedman, M. Osborne, A. Sarkar, S. Clark, R. Hwa, J. Hockenmaier, P. Ruhlen, S. Baker, and J. Crim. Bootstrapping statistical parsers from small data sets. In *Proceedings of the 10th Conference on the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, pp. 331-338, 2003.

[Syswerda 1989] Syswerda. Uniform crossover in genetic algorithms. *Proceedings of the 3rd International Conference on Genetic Algorithms*, pp. 2-9, 1989.

[Szummer 2002] M. Szummer and T. Jaakkola. Information regularization with partially labeled data. *Advances in Neural Information Processing Systems, 15*, 2002.

[Tan 2005] P-N. Tan, M. Steinbach and V. Kumar. Introduction to Data Mining. Addison-Wesley, ISBN: 0321321367, 2005.

[Terabe 2008] M. Terabe and K. Hashimoto. Evaluation Criteria of Feature Splits for Co–Training. In *Proceedings of the International MultiConference of Engineers and Computer Scientists 2008,* 2008.

[Ting 2011] S.L. Ting, W.H. Ip and Albert H.C. Tsang. Is Naïve Bayes a Good Classifier for Document Classification. International Journal of Software Engineering and Its Applications Vol. 5, No. 3, July, 2011.

[Turney 2002] P. Turney. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the Association for Computational Linguistics (ACL)*, pp. 417–424, 2002.

[Yang 1997] Y. Yang and J. O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In D. H. F. Jr., ed., *Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997)*, pp. 412-420, Morgan Kaufmann, Nashville, Tennessee, USA, 1997.

[Yarowsky 1995] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics,* pp. 189–196, 1995

[Zhang 2004] Zhang, L., Zhu, J., and Yao, T. 2004. An evaluation of statistical spam filtering techniques. *ACM Transactions on Asian Language Information Processing (TA LIP)* 3(4): 243–269.

[Zhang 2011] P. Zhang and Z. Obradovic. Learning from Inconsistent and Unreliable Annotators by a Gaussian Mixture Model and Bayesian Information Criteria. *In Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 553-568, 2011.

[Zhang 2014] Y. Zhang, J. Wen, X. Wang and Z. Jiang. Semi-supervised learning combining co-training with active learning. *Expert Systems with Applications*, Vol. 41, Issue 5, pp. 2372-2378, ISSN 0957-4174, 2014.

[Zhou 2004] Y. Zhou and S. Goldman. Democratic co-learing. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*, 2004.

[Zhou 2005a] Z.-H. Zhou and Ming Li. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Trans. Knowledge and Data Engineering* 17, pp. 1529-1541, 2005.

[Zhou 2005b] Z.-H. Zhou and M. Li. Semi-supervised regression with co-training. *International Joint Conference on Artificial Intelligence (IJCAI)*, 2005.

[Zhou 2007a] Z.-H. Zhou and J.-M. Xu. On the relation between multi-instance learning and semi-supervised learning. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[Zhou 2007b] Z.-H. Zhou, D.-C. Zhan, and Q. Yang. Semi-supervised learning with very few labeled training examples. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, pp. 675–680, Vancouver, Canada, 2007.

[Zhou 2009] Z–H. Zhou. When semi–supervised learning meets ensemble learning. *In Proceedings of International Workshop on Multiple Classifier Systems*, pp. 529–538, 2009.

[Zhu 2008] X. Zhu, "Semi-Supervised Learning Literature Survey", Technical Report Computer Sciences TR 1530 University of Wisconsin Madison. Last modified on July 19, 2008.

[Vapnik 1963] V.N. Vapnik. The Nature of Statistical Learning Theory, Springer-Verlag, New York, 1963.

[Wan 2009] Wan, X., 2009. Co-training for cross-lingual sentiment classification. In: ACL/AFNLP, pp. 235–243

[Wang 2007] W. Wang and Z.-H. Zhou. Analyzing co-training style algorithms. In *Proceedings of the 18th European Conference on Machine Learning*, pp. 454–465, Warsaw, Poland, 2007.

[Wang 2012] Y. Wang, S.C. Chan, and G. Ngai. Applicability of Demographic Recommender System to Tourist Attractions: A Case Study on Trip Advisor. *Web Intelligence/IAT Workshops* pp. 97-101, 2012.

[Wiebe 2001] J. Wiebe, R. Bruce, M. Bell, M. Martin and T. Wilson. A Corpus Study of Evaluative and Speculative Language. In *Proceedings of the 2nd ACL SIG on Dialogue Workshop on Discourse and Dialogue*. Aalborg, Denmark 2001.

[Wiebe 2005] J. Wiebe and E. Riloff. Creating subjective and objective sentence classifiers from unannotated texts. In *Proceedings of the Conference on Computational Linguistics and Intelligent Text Processing (CICLing),* volume 3406, pages 486–497. Springer, 2005.

[Wright 1992] S.P. Wright Adjusted P-values for simultaneous inference. Biometrics 48, pp. 1005–1013, 1992.

Biografija

Jelena Slivka rođena je 01.02.1985. godine u Novom Sadu, republika Srbija. Matematičko odeljenje gimnazije „Jovan Jovanović Zmaj" završila je u Novom Sadu sa odličnim uspehom. Studije na Fakultetu tehničkih nauka u Novom Sadu upisala je 2003. godine. Završila je 2008. godine Integrisane osnovne i diplomske akademske – Master studije na studijskom programu Računarstvo i automatika – Primenjene računarske nauke i informatika sa prosečnom ocenom 9.76 u toku studija i postignutim ukupnim brojem ESPB bodova 304. Diplomski – master rad na temu „Editor modela elektroenergetskih objekata za DMS softver" odbranila je sa ocenom 10. Od 2003. do 2008. godine bila je stipendista Ministarstva prosvete i sporta Republike Srbije i stipendista kompanije „DMS group", a od 2009. do 2011. godine bila je stipendista Ministarstva za nauku i tehnološki razvoj. Dobila je nagradu Univerziteta u Novom Sadu za postignut uspeh u školskim godinama 2003/2004, 2005/2006, 2006/2007 i 2007/2008. Od školske 2008/2009 godine je student doktorskih studija na Fakultetu tehničkih nauka – smer Računarstvo i automatika, usmerenje Računarske nauke i informatika. Položila je sve ispite predviđene planom i programom studijskog programa Računarstvo i automatika sa prosečnom ocenom 10. U oktobru 2009. izabrana je u zvanje saradnika u nastavi za užu naučnu oblast Računarske nauke i informatika na Fakultetu tehničkih nauka Univerziteta u Novom Sadu. Kasnije, 01.10.2011. godine je izabrana u zvanje asistenta za užu naučnu oblast Primenjene računarske nauke i informatika na istom fakultetu. Autor je 11 publikovanih naučnih radova. Učestvovala je u izradi više stručnih i naučnih projekata.

Živi u Novom Sadu. Od stranih jezika govori engleski jezik.

UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
Ključna dokumentacijska informacija

Redni broj: **RBR**	
Identifikacioni broj: **IBR**	
Tip dokumentacije: **TD**	Monografska dokumentacija
Tip zapisa: **TZ**	Tekstualni štampani materijal
Vrsta rada: **VR**	Doktorska disertacija
Autor: **AU**	Jelena Slivka
Mentor: **MN**	dr Aleksandar Kovačević, docent, Fakultet Tehničkih Nauka, Novi Sad
Naslov rada: **NR**	Adaptivni sistem za automatsku polu-nadgledanu klasifikaciju podataka
Jezik publikacije: **JP**	srpski (latinica)
Jezik izvoda: **JI**	srpski (latinica) / engleski
Zemlja publikovanja: **ZP**	Republika Srbija
Uže geografsko područje: **UGP**	Vojvodina
Godina: **GO**	2014.
Izdavač: **IZ**	Autorski reprint
Mesto i adresa: **MA**	Fakultet Tehničkih Nauka, Trg Dositeja Obradovića 6, Novi Sad, Republika Srbija
Fizički opis rada: **FO**	(8/168/152/20/14/27/0) (broj poglavlja/strana/lit.citata/ tabela/slika/grafika/priloga)
Naučna oblast: **NO**	Informatika

Naučna disciplina: **ND**	Istraživanje podataka
Predmetna odrednica/ ključne reči: **PO** **UDK**	Istraživanje podataka, klasifikacija, polu-nadgledano obučavanje, ko-trening, tehnike učenja sa grupom hipoteza
Čuva se: **ČU**	Biblioteka Fakulteta tehničkih nauka, Trg Dositeja Obradovića 6, Novi Sad
Važna napomena: **VN**	Nema
Izvod: **IZ**	**Cilj –** Cilj istraživanja u okviru doktorske disertacije je razvoj sistema za automatsku polu-nadgledanu klasifikaciju podataka. Sistem bi trebao biti primenljiv na širokom spektru domena gde je neophodna klasifikacija podataka, a teško je, ili čak nemoguće, doći do dovoljno velikog i raznovrsnog obučavajućeg skupa podataka **Metodologija –** Modeli opisani u disertaciji se baziraju na kombinaciji ko-trening algoritma i tehnika učenja sa grupom hipoteza. Prvi korak jeste obučavanje grupe klasifikatora velike raznolikosti i kvaliteta. Sa ovim ciljem modeli eksploatišu primenu različitih konfiguracija ko-trening algoritma na isti skup podataka. Prednost ovog pristupa je mogućnost korišćenja značajno manjeg anotiranog obučavajućeg skupa za inicijalizaciju algoritma. Skup nezavisno obučenih ko-trening klasifikatora se kreira generisanjem predefinisanog broja slučajnih podela obeležja polaznog skupa podataka. Nakon toga se, polazeći od istog inicijalnog obučavajućeg skupa, ali korišćenjem različitih kreiranih podela obeležja, obučava grupa ko-trening klasifikatora. Nakon ovoga, neophodno je kombinovati predikcije nezavisno obučenih klasifikatora. Predviđena su dva načina kombinovanja predikcija. Prvi način se zasniva na klasifikaciji zapisa na osnovu većine glasova grupe ko-trening klasifikatora. Na ovaj način se daje predikcija za svaki od zapisa koji su pripadali grupi neanotiranih primera korišćenih u toku obuke ko-treninga. Potom se primenjuje genetski algoritam u svrhu selekcije najpouzdanije klasifikovanih zapisa ovog skupa. Konačno,

	najpouzdanije klasifikovani zapisi se koriste za obuku finalnog klasifikatora. Ovaj finalni klasifikator se koristi za predikciju klase zapisa koje je neophodno klasifikovati. Opisani algoritam je nazvan Algoritam Statistike Slučajnih Podela (*Random Split Statistics algorithm, RSSalg*). Drugi način kombinovanja nezavisno obučenih ko-trening klasifikatora se zasniva na *GMM-MAPML* tehnici estimacije tačnih klasnih obeležja na osnovu višestrukih obeležja pripisanih od strane različitih anotatora nepoznatog kvaliteta. U ovom algoritmu, nazvanom Integracija Višestrukih Ko-treninranih Klasifikatora (Integration of Multiple Co-trained Classifiers, IMCC), svaki od nezavisno treniranih ko-trening klasifikatora daje predikciju klase za svaki od zapisa koji je neophodno klasifikovati. U ovoj postavci se svaki od ko-trening klasifikatora tretira kao jedan od anotatora čiji je kvalitet nepoznat, a svakom zapisu, za koga je neophodno odrediti klasno obeležje, se dodeljuje više klasnih obeležja. Na kraju se primenjuje *GMM-MAPML* tehnika, kako bi se na osnovu dodeljenih višestrukih klasnih obeležja za svaki od zapisa izvršila estimacija stvarnog klasnog obeležja zapisa. **Rezultati –** U disertaciji su razvijena dva modela, Integracija Višestrukih Ko-treninranih Klasifikatora (IMCC) i Algoritam Statistike Slučajnih Podela (RSSalg), bazirana na ko-trening algoritmu, koja rešavaju zadatak automatske klasifikacije u slučaju nepostojanja dovoljno velikog anotiranog korpusa za obuku. Modeli predstavljeni u disertaciji dizajnirani su tako da omogućavaju primenu ko-trening algoritma na skupove podataka bez prirodne podele obeležja, kao i da unaprede njegove performanse. Modeli su na više skupova podataka različite veličine, dimenzionalnosti i redudantnosti poređeni sa postojećim ko-trening alternativama. Pokazano je da razvijeni modeli na testiranim skupovima podataka postižu bolje performanse od testiranih ko-trening alternativa. **Praktična primena –** Razvijeni modeli imaju široku mogućnost primene u svim domenima gde je neophodna klasifikacija podataka, a anotiranje podataka dugotrajno i skupo. U disertaciji je prikazana i primena razvijenih modela u nekoliko konkretnih

	situacija gde su modeli od posebne koristi: detekcija subjektivnosti, više-kategorijska klasifikacija i sistemi za davanje preporuka. **Vrednost –** Razvijeni modeli su korisni u širokom spektru domena gde je neophodna klasifikacija podataka, a anotiranje podataka dugotrajno i skupo. Njihovom primenom se u značajnoj meri smanjuje ljudski rad neophodan za anotiranje velikih skupova podataka. Pokazano je da performanse razvijenih modela prevazilaze performanse postojećih alternativa razvijenih sa istim ciljem relaksacije problema dugotrajne i mukotrpne anotacije velikih skupova podataka.
Datum prihvatanja teme od NN veća: **DP**	29.05.2014.
Datum odbrane: **DO**	
Članovi komisije: **KO**	
Predsednik:	dr Surla Dušan, profesor emeritus Prirodno-matematičkog fakulteta, Univerziteta u Novom Sadu
član:	dr Konjović Zora, redovni profesor Fakulteta tehničkih nauka, Univerziteta u Novom Sadu
član:	dr Milosavljević Milan, redovni profesor Elektrotehničkog fakulteta, Univerziteta u Beogradu
član:	dr Ćulibrk Dubravko, vanredni profesor profesor Fakulteta tehničkih nauka, Univerziteta u Novom Sadu
član:	dr Malbaša Vuk, docent Fakulteta tehničkih nauka, Univerziteta u Novom Sadu
Član, mentor:	dr Aleksandar Kovačević, docent Fakulteta tehničkih nauka, Univerziteta u Novom Sadu, mentor

UNIVERSITY OF NOVI SAD
FACULTY OF SCIENCE

Key words documentation

Accession number: **ANO**	
Identification number: **INO**	
Document type: **DT**	Monograph publication
Type of record: **TR**	Printed text
Content code: **CC**	PhD Thesis
Author: **AU**	Jelena Slivka
Mentor/comentor: **MN**	Aleksandar Kovačević, PhD, assistant professor, Faculty of Technical Sciences, Novi Sad
Title: **TI**	Adaptive System for Automated Semi-supervised Data Classification
Language of text: **LT**	Serbian (Latin)
Language of abstract: **LA**	Serbian (Latin) / English
Country of publication: **CP**	Serbia
Locality of publication: **LP**	Vojvodina
Publication year: **PY**	2014
Publisher: **PU**	Author's reprint
Publication place: **PP**	Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad
Physical description: 1.4.3.3.3 **PD**	(8/168/152/20/14/27/0) (chapters/pages/literature/tables/ pictures/graphs/appendix)
Scientific field: **SF**	Informatics

Scientific discipline: **SD**	Data mining
Subject/ Key words: **SKW** **UC**	Data mining, classification, semi-supervised learning, co-training, ensemble learning
Holding data: **HD**	Library of Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad
Note: **N**	None
Abstract: **AB**	**Aim –** The research presented in this thesis is aimed towards the development of the system for automatic semi-supervised classification. The system is designed to be applicable on the broad spectrum of practical domains where automatic classification of data is needed but it is hard or impossible to obtain a large enough training set. **Methodology** – The described models combine co-training algorithm with ensemble learning with the aim to overcome the problem of co-training application on the datasets without the natural feature split. The first step is to create the ensemble of co-training classifiers. For this purpose the models presented in this thesis apply different configurations of co-training on the same training set. Compared to existing similar approaches, this approach requires a significantly smaller initial training set. The ensemble of independently trained co-training classifiers is created by generating a predefined number of random feature splits of the initial training set. Using the same initial training set, but different feature splits, a group of co-training classifiers is trained. The two models differ in the way the predictions of different co-training classifiers are combined. The first approach is based on majority voting: each instance recorded in the enlarged training sets resulting from co-training application is classified by majority voting of the group of obtained co-training classifiers. After this, the genetic algorithm is applied in order to select the group of most reliably classified instances from this set. The most reliable instances are used in

	order to train a final classifier which is used to classify new instances. The described algorithm is called Random Split Statistic Algorithm (*RSSalg*). The other approach of combining single predictions of the group of co-training classifiers is based on *GMM-MAPML* technique of estimating the true hidden label based on the multiple labels assigned by multiple annotators of unknown quality. In this model, called the Integration of Multiple Co-trained Classifiers (*IMCC*), each of the independently trained co-training classifiers predicts the label for each test instance. Each co-training classifier is treated as one of the annotators of unknown quality and each test instance is assigned multiple labels (one by each of the classifiers). Finally, *GMM-MAPML* technique is applied in order to estimate the true hidden label in the multi-annotator setting. **Results** – In the dissertation the two models are developed: the Integration of Multiple Co-trained Classifiers (*IMCC*) and Random Split Statistic Algorithm (*RSSalg*). The models are based on co-training and aimed towards enabling automatic classification in the cases where the existing training set is insufficient for training a quality classification model. The models are designed to enable the application of co-training algorithm on datasets that lack the natural feature split needed for its application, as well as with the goal to improve co-training performance. The models are compared to their co-training alternatives on multiple datasets of different size, dimensionality and feature redundancy. It is shown that the developed models exhibit superior performance compared to considered co-training alternatives. **Practical application** – The developed models are applicable on the wide spectrum of domains where there is a need for automatic classification and training data is insufficient. The dissertation presents the successful application of models in several concrete situations where they are highly

	beneficial: subjectivity detection, multicategory classification and recommender systems. **Value** – The models can greatly reduce the human effort needed for long and tedious annotation of large datasets. The conducted experiments show that the developed models are superior to considered alternatives.
Accepted by the Scientific Board: **ASB**	29.05.2014.
Defended on: **DE**	
Thesis defend board: **DB**	
President:	Surla Dušan, PhD, professor emeritus, Faculty of sciences, University of Novi Sad
Member:	Konjović Zora, PhD, full professor, Faculty of Technical Sciences,University of Novi Sad
Member:	Milosavljević Milan, full profesor, Faculty of Electrical Engineering and Computer Science,University of Belgrade
Member:	Ćulibrk Dubravko, PhD, associate profesor, Faculty of Technical Sciences,University of Novi Sad
Member:	Malbaša Vuk, PhD, assistant profesor, Faculty of Technical Sciences,University of Novi Sad
Member, Mentor:	Aleksandar Kovačević, PhD, assistant professor, Faculty of Technical Sciences,University of Novi Sad, advisor