UNIVERZITET SINGIDUNUM BEOGRAD

DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE

NOVA METODA ZA KLASTEROVANJE LJUDSKIH AKTIVNOSTI BAZIRANA NA GRAFOVIMA

DOKTORSKA DISERTACIJA

Mentor:

prof. dr Nebojša Bačanin-Džakula

Kandidat:

Nebojša Budimirović

Beograd, 2021.

SINGIDUNUM UNIVERSITY BELGRADE

DEPARTMENT FOR POSTGRADUATE STUDIES

A NOVEL GRAPH-BASED METHOD FOR CLUSTERING HUMAN ACTIVITIES

PhD Thesis

Mentor:

Professor Nebojša Bačanin-Džakula

Candidate:

Nebojša Budimirović

Belgrade, 2021.

MENTOR:

Prof. dr Nebojša Bačanin-Džakula

Univerzitet Singidunum, Beograd

ČLANOVI KOMISIJE:

Prof. dr Miodrag Živković

Univerzitet Singidunum, Beograd

Prof. dr Boško Nikolić

Univerzitet u Beogradu, Elektrotehnički fakultet, Beograd

Želim da izrazim veliku zahvalnost profesoru dr Nebojši Bačaninu na stručnoj pomoći i ohrabrenjima koja su me podsticala da istrajem u radu.

Posebno se zahvaljujem porodici na nesebičnoj pomoći i podršci.

Sadržaj

1 Uvod

1.1 Predmet i problem istraživanja

Predmet istraživanja ove disertacije su metode klasterovanja koje su pogodne za prepoznavanje složenih ljudskih aktivnosti. Problem istraživanja glasi: Da li je moguće postizanje boljih rezultata, od onih koji se dobijaju postojećim metodama, za prepoznavanje složenih ljudskih aktivnosti u realnom okruženju?

Automatski razvoj prepoznavanja ljudskih aktivnosti (eng. human activity recognition - HAR) izazovni je zadatak koji tek treba rešiti. Rešenje ovog zadatka ima veliki značaj za unapređenje interakcije čovek-mašina, sigurnosti, zdravstvene zaštite i mnogih drugih oblasti. Kako računarski hardver postaje sve pristupačniji, manji i brži, on postaje i sveprisutan, toliko da je danas pametni telefon, pametni sat ili sličan uređaj preuzeo integralnu ulogu u svakodnevnom životu.

Porast takvih nosivih uređaja opremljenih inercijalnim mernim jedinicama (eng. inertial measurement units - IMU) doveo je do povećanja obima korišćenja podataka sa IMU za HAR. U prilog toj činjenici ide veliki broj objavljenih radova u eminentnim međunarodnim časopisima koji se nalaze na Thomson Reuters SCI listama, kao i na uglednim međunarodnim konferencijama koje organizuju IEEE i ACM. Primena različitih metoda, koristeći IMU za HAR, daje zapažene rezultate na polju automatskog prepoznavanja na konvencionalnim skupovima podataka HAR, koji sadrže isključivo jednostavne i ponavljajuće aktivnosti.

Međutim, javlja se potreba za analizom složenih aktivnosti koje nisu precizno definisane. Razni ljudski subjekti istu aktivnost obavljaju na različite načine. Segmenti iste složene aktivnosti mogu se izvoditi različitim redosledom. Pored toga, različite vrste složenih aktivnosti mogu imati iste segmente. Algoritmi klasterovanja grafova mogu biti praktični za prepoznavanje i klasifikaciju ljudskih aktivnosti.

1.2 Cilj i zadaci istraživanja

Cilj istraživanja u doktorskoj disertaciji je konstrukcija pristupa koji omogućava unapređenje rešavanja problema prepoznavanja i klasterovanja složenih ljudskih aktivnosti u realnom okruženju.

Da bi se realizovao cilj, potrebno ga je konkretizovati kroz zadatke koji predstavljaju uže ciljeve:

1. Predstavljanje ljudskih aktivnosti simboličkim modeliranjem prostorno-vremenskih signala sa nosivih uređaja.
2. Određivanje adekvatne mere sličnosti podataka dobijenih modeliranjem ljudskih aktivnosti.
3. Konstruisanje algoritma klasterovanja grafa pogodnog za klasterovanje složenih ljudskih aktivnosti u slučaju kada broj klastera nije unapred poznat.
4. Konstruisanje algoritma klasterovanja grafa pogodnog za klasterovanje složenih ljudskih aktivnosti u slučaju kada je broj klastera unapred dat.
5. Predstavljanje nove metode evaluacije za precizniju ocenu performansi predloženog pristupa.

1.3 Hipoteze istraživanja

Na osnovu formulisanog cilja i zadataka istraživanja, formulišu se hipoteze pomoću kojih se dobija odgovor na pitanje postavljeno kao problem istraživanja.

Opšta hipoteza u istraživanju za potrebe izrade doktorske disertacije može se formulisati na sledeći način: "Implementacijom novog pristupa, moguće je unaprediti rešavanje problema prepoznavanja i klasterovanja složenih ljudskih aktivnosti u realnom okruženju".

Pojedinačne hipoteze su:

1. Složene ljudske aktivnosti se mogu adekvatno predstaviti stringovima, dobijenim simboličkim modeliranjem prostorno-vremenskih signala sa nosivih

uređaja.

2. Adaptacijom neke od postojećih mera sličnosti između stringova može se dobiti adekvatna mera sličnosti ljudskih aktivnosti.

3. Može se konstruisati algoritam za klasterovanje težinskih grafova pogodan za klasterovanje složenih ljudskih aktivnosti, kada broj klastera nije unapred poznat.

4. Može se konstruisati algoritam za klasterovanje težinskih grafova pogodan za klasterovanje složenih ljudskih aktivnosti, kada je broj klastera unapred dat.

5. Moguće je novom metodom evaluacije unaprediti analizu rezultata klasterovanja.

Sve hipoteze su potvrđene.

1.4 Metode istraživanja

Da bi se razvile hipoteze koje su u skladu sa predmetom i ciljem istraživanja, primenjene su metode istraživanja koje su uobičajene u praksi i naučno potvrđene: analitičko−sintetička metoda, deduktivno−induktivna metoda, komparativno-kvantitativna analiza, empirijska metoda, evaluaciona metoda, modeliranje i eksperimentalni deo za dokazivanje hipoteza.

Analitičkom metodom je urađena analiza naučnih i stručnih radova iz ove oblasti, objavljenih u relevantnim časopisima i drugim elektronskim i štampanim publikacijama. Sintetičkom metodom su integrisani pojedinačni aspekti posmatrane pojave, dobijeni prethodnom analizom, u svrhu opisivanja određenih pravila i strukture pojave koja je predmet ovog istraživanja. Drugim rečima, svi prikupljeni podaci su grupisani metodom sinteze, kako bi se došlo do pouzdanih zaključaka.

Korišćenje deduktivno - induktivne metode je u funkciji usmeravanja istraživanja od generalnog ka posebnom, odnosno od posebnog ka generalnom

da bi se dobili odgovarajući zaključci.

U okviru komparativno-kvantitativne analize, a u korelaciji sa ciljevima istraživanja i pojavom koja je predmet istraživanja, kao najprikladniji postupak u istraživanju korišćena je metoda analize. Da bi se analizirali dobijeni rezultati primenom odgovarajućih metoda, tehnika, kao i algoritama klasterovanja u svrhu uspešnog prepoznavanja i klasterovanja složenih ljudskih aktivnosti u realnom okruženju, primenjen je postupak kvantitativne analize. Za upoređivanje rezultata dobijenih novim pristupom za prepoznavanje ljudskih aktivnosti sa rezultatima dobijenim drugim poznatim pristupima korišćena je metoda komparativne analize. Izveden je zaključak o stvarnom unapređenju rešavanja ovog problema primenom novog pristupa i posebno novih algoritama klasterovanja.

Primenom empirijske metode prikupljeni primarni podaci su razmatrani i u svetlu diskutovanih postojećih teorijskih i empirijskih nalaza, odnosno sekundarnih izvora podataka, kako bi se testirale polazne istraživačke hipoteze. Praktičnim eksperimentisanjem (primenom algoritma) došlo se do primarnih podataka.

Pod modeliranjem se podrazumeva konstrukcija matematičkih modela koji predstavljaju validnu predstavu problema koji se izučava i rešava. Za evaluaciju rezultata klasterovanja, pored poznatih, korišćena je i nova metoda evaluacije.

1.5 Glavni naučni doprinosi

Glavni motiv istraživanja, koje je sprovedeno za potrebe doktorske disertacije, je realno očekivanje da će novi pristup za rešavanje problema prepoznavanja složenih ljudskih aktivnosti dati bolje rezultate od već postojećih i prikazanih metoda, algoritama i tehnika za rešavanje istog tipa problema. Na osnovu sprovedenog istraživanja mogu da se izdvoje sledeći naučni doprinosi:

- Novi algoritam za klasterovanje težinskih grafova, pogodan za klasterovanje složenih ljudksih aktivnosti, kada broj klastera nije unapred poznat.
- Novi algoritam za klasterovanje težinskih grafova, pogodan za klasterovanje

složenih ljudksih aktivnosti, kada je broj klastera unapred dat.

- Unapređenje tehnike obrade signala sa prenosivih uređaja za potrebe analize ljudskih aktivnosti.

- Nova metoda evaluacije rezultata klasterovanja.

1.6 Okvirni sadržaj doktorske disertacije

Disertacija se sastoji iz uvoda, šest poglavlja i zaključka.

- U uvodnom delu doktorske disertacije ukratko je izložen problem koji je razmatran, kao i pregled dosadašnjih rezultata iz ove oblasti. Takođe je izložen metodološki pristup, kao i struktura rada. Na kraju su izloženi najvažniji doprinosi rada.

- U oviru drugog poglavlja je izložena taksonomija pristupa za analizu prepoznavanja ljudskih aktivnosti. Posebno je prikazana klasifikacija ljudskih aktivnosti i tipovi senzora. Zatim se daju pregled baza podataka i vrste zadataka analize ljudskih aktivnosti.

- U trećem poglavlju su izložene osnove klaster analize. Takođe se daje klasifikacija i karakteristike metoda, kao i pregled algoritama klasterovanja.

- U četvrtom poglavlju su predstavljeni novi algoritmi za klasterovanje težinskih grafova i obrazložena je njihova pogodnost za primenu u prepoznavanju složenih ljudskih aktivnosti.

- U petom poglavlju je detaljno opisana obrada signala i odabir parametara koji su neophodni da bi se definisale karakteristike ili sličnosti koje se analiziraju. Posebno je predstavljena nova tehnika transformacije signala za ekstraktovanje potrebnih parametara za klasterovanje, kao i nova mera sličnosti.

- Šesto poglavlje sadrži pregled metoda evaluacije rezultata klasterovanja. U svrhu preciznije ocene rezultata klasterovanja predložena je i nova metoda

evaluacije. Takođe su diskutovane njene prednosti u odnosu na slične metode evaluacije.

- U okviru sedmog, eksperimentalnog, poglavlja detaljno su prikazane implementacije predloženog pristupa, odnosno formiranje i klasterovanje težinskih grafova. U tu svrhu su korišćene različite baze podataka dobijene sa uređaja kojima su beležene ljudske aktivnosti. Pored novih algoritama, radi komparativne analize, na istim uzorcima su testirani i algoritmi drugih autora. Takođe je prikazana evaulacija i diskusija rezultata klasterovanja.

Na kraju disertacije je iznet zaključak sa mogućim pravcima daljeg istraživanja u ovoj perspektivnoj oblasti.

2 Prepoznavanje ljudskih aktivnosti

Automatsko prepoznavanje ljudskih aktivnosti je veoma aktuelna tema istraživanja zbog mnogobrojnih primena u raznim poljima, kao što su interakcija čovek-mašina, praćenje zdravlja i rehabilitacija, video nadzor, sport, kompjuterske igre itd.

2.1 Taksonomija

HAR se može definisati kao sposobnost prepoznavanja ili detektovanja trenutnih aktivnosti pomoću informacija prikupljenih sa raznih senzora. Senzori mogu biti nosivi ili prikačeni na objekte u svakodnevnoj upotrebi, kamere, senzori postavljeni u okruženju. Napretkom tehnologije i sve pristupačnijim cenama uređaja, praćenje dnevnih aktivnosti postalo je praktično i popularno. Razni pristupi su korišćeni da bi se zabeležile takve aktivnosti. Te pristupe možemo podeliti u dve kategorije: vizuelno-bazirane i tehnike bazirane na senzorima. Vizuelno-bazirane tehnike se služe kamerama da bi zabeležile ljudske aktivnosti i promene u okolini. Iako su lake za korišćenje i pružaju dobre rezultate, imaju nedostataka od kojih je krucijalni privatnost. Usled napretka u tehnologiji i pristupačnosti, istraživanje u ovom polju uglavnom naginje ka metodama baziranim na senzorima. Tehnike bazirane na senzorima koriste mrežu senzora i povezanih uređaja da bi zabeležile ljudske aktivnosti. Ove tehnike možemo podeliti u tri katagorije na osnovu položaja senzora: tehnike sa nosivim senzorima, senzorima na objektima i ambijentalnim senzorima. Nosive senzore postavljamo na određene delove tela (članak, lakat, koleno, šaka itd.) osobe da bi se beležile njene aktivnosti. Senzori na objektima se postavljaju na objekte sa kojima osobe dolaze u kontakt (šporet, telefon, sofa, krevet, frižider itd.). Ambijentalni senzori su implementirani u okruženje i omogućavaju beleženje različitih promena (osvetljenje, temperatura i sl.) u određenom prostoru.

Prepoznavanje aktivnosti u pametnim okruženjima nailazi na brojne izazove. Individue često izvršavaju dnevne aktivnosti različito, koristeći visok stepen slobode u redosledu i trajanju radnji koje obavljaju. Ponekad je potrebno spojiti

i interpretirati podatke sa senzora iz više izvora da bi se uspostavio kontekst svakodnevnih aktivnosti.

Još jedna podela prepoznavanja ljudskih aktivnosti je na pristupe bazirane na prikupljanju podataka i pristupe bazirane na znanju. Pristupi bazirani na prikupljanju podataka uče aktivnosti iz postojećih obimnih baza podataka koristeći tehnike istraživanja podataka i mašinskog učenja. Prednost takvih tehnika je omogućavanje rukovanja nesigurnim, nepotpunim i vremenskim informacijama. Ove tehnike omogućavaju visok nivo prepoznavanja i prilagodljivosti novim situacijama. Ipak, nedostatak istih se ogleda u potrebi za velikom količinom podataka, skalabilnosti i ponovne upotrebe. Dobre performanse pružaju samo kada su im date dobro dizajnirane ulazne karakteristike. Nasuprot tome, pristupi bazirani na znanju grade modele aktivnosti eksploatišući prethodno opsežno znanje. Takvi pristupi su motivisani opažanjem realnog okruženja koje uključuje svakodnevne aktivnosti i liste predmeta potrebnih za izvršavanje tih aktivnosti. Struktura znanja modelirana je i predstavljena kroz forme, kao što su šeme, pravila ili mreže. Dok su prednosti takvih modela jasnost, logičnost i lakoća inicijalizovanja, mana je nemogućnost rukovanja nesigurnim i vremenskim informacijama i prilagođavanje promenama i novim okruženjima. Pristupi bazirani na prikupljanju podataka zahtevaju više podataka i komputacionog vremena od pristupa baziranih na znanju, ali sve veći broj baza podataka i veća komputaciona snaga umanjuju ove teškoće i omogućavaju treniranje kompleksnijih modela, što može da prevaziđe zavisnost od ulaznih karakteristika.

Mnoge baze podataka su struktuirane radi analiziranja ljudskih aktivnosti. Baza podataka može biti generisana iz realnog okruženja ili može biti sintetički generisana specijalizovanim softverom. Takođe, baza podataka može biti obeležena kada subjekti prijave aktivnosti koje obavljaju, vođenjem evidencije ili pomoću uređaja, odnosno, neobeležena kada su ove informacije nedostupne.

Različiti zadaci mašinskog učenja se koriste za analizu prepoznavanja ljudskih aktivnosti. Zadatke mašinskog učenja, koji mogu biti korišćeni za ekstrakciju informacija iz baza podataka, možemo podeliti na: klasifikaciju, regresiju i klasterovanje. Klasifikacija omogućava identifikaciju predefinisane klase kojoj

data činjenica pripada. Regresija se koristi za pronalaženje numeričke vrednosti neprekidne varijable. Klasterovanje omugućava grupisanje objekata po njihovoj sličnosti.

Prepoznavanje ljudskih aktivnosti može biti analizirano u dva scenarija, nadgledanom i nenadgledanom. U nadgledanom scenariju, informacije o kriterijumima klasa su dostupne. Nasuprot tome, u nenadgledanom scenariju su te informacije nedostupne. Nadgledani scenario zahteva podelu skupa podataka na skup za treniranje i skup za testiranje. Nenadgledani scenario ne zahteva podatke za treniranje.

Aktivnosti zabeležene pomoću senzora mogu biti: jednokorisničke, isprekidane, višekorisničke i uporedne. Aktivnost je jednokorisnička kada jedna osoba vrši razne aktivnosti u datom vremenu, s tim da pre svake nove radnje, prethodna mora biti završena. Aktivnost je isprekidana ako je zabeležena kroz interakciju jedne osobe u određenom prostoru, s tim da osoba može započeti novu aktivnost bez da je završila prethodno započetu. Aktivnost je višekorisnička kada je zabeležena kroz interakciju više osoba u određenom prostoru, koje mogu izvršavati razne aktivnosti simultano bez da su završili sa prethodno započetim aktivnostima. Aktivnost je uporedna kada osoba obavlja više aktivnosti u isto vreme.

Tehnike prepoznavanja ljudskih aktivnosti mogu biti podeljene u tri kategorije: tehnike bazirane na akciji, na interakciji i na pokretima. Ovo istraživanje bavi se aktivnostima baziranim na akciji, a one se dele na: prepoznavanje gestikulacija, položaja tela, ponašanja, svakodnevnih aktivnosti, detekcije pada i aktivnosti uz pomoć ambijenta. Prepoznavanje gestikulacije je tehnika koja koristi senzore da prepozna i interpretira pokrete šake, ruke ili nekog drugog dela tela kao komande (npr. mahanje, tapšanje). Prepoznavanje položaja tela predstavlja pozicioniranje subjekta u određenu pozu (stajanje, sedenje, lezanje, čučanje itd.). Prepoznavanje ponašanja svodi se na uočavanje ponašanja osobe pomoću podataka sa različitih senzora. Prepoznavanje svakodnevnih aktivnosti podrazumeva identifikovanje uobičajenih fundamentalnih aktivnosti (hodanje, trčanje, spavanje, penjanje uz i silaženje niz stepenice), kao i složenih aktivnosti (pripre-

manje jela, kupanje, oblačenje, pranje sudova), koje se obavljaju na dnevnoj bazi. Detekcija pada i prepoznavanje aktivnosti uz pomoć ambijenta podrazumevaju korišćenje ambijentalne inteligentne tehnike i procesa koji omogućavaju starijim osobama da žive nezavisno.

Senzori korišćeni u ovom istraživanju opisani su u sledećoj podsekciji.

2.2 Navigacioni senzorski sistem

Ranije su navigacioni sistemi korišćeni uglavnom u vazduhoplovstvu, plovidbi i kosmonautici. Zbog svojih dimenzija nisu bili prikladni za beleženje podataka o ljudskim aktivnostima. Savremeni navigacioni sistemi imaju značajno manje dimenzije i dosta su jednostavniji za upotrebu, pa je time stvorena mogućnost da se primene prilikom praćenja ljudskih aktivnosti. To je omogućeno, pre svega razvojem nove, tzv. MEMS (engl. Micro Electro Mechanical Systems) tehnologije. MEMS senzori se uglavnom ugrađuju u jedno kućište, tako da su veoma pogodni za praktičnu upotrebu.

Merni uređaji kojima se dobijaju informacije o ljudskim aktivnostima nazivaju se senzori. Senzori nove MEMS tehnologije su relativno malih dimenzija, laki i jeftini. Ovi senzori su sastavni delovi merne jedinice MIMU (engl. Magnetic Inertial Measurement Unit). U okviru merne jedinice postoje inercijalni i magnetni senzori, a to su: žiroskopi, akcelerometri i magnetometri. Svaki senzor beleži signale duž svake od koordinatnih osa. Pomoću akcelerometra se dobijaju podaci o translatornom, a pomoću žiroskopa o rotacionom kretanju.

Žiroskopom se dobija vrednost brzine rotacije oko senzorskih osa, odnosno ugaona brzina. Vrednosti ugaone brzine su izražene u radijanima u sekundi (rad/s). Akcelerometar meri ubrzanje (izraženo preko gravitacionog ubrzanja g duž koordinatnih osa x, y, z. Geomagnetski senzor (magnetometar) daje podatke orijentacije senzora u odnosu na geomagnetsko polje (izražene u stepenima). Pomoću ova tri senzora dobija se 9 informacija koje određuju položaj tela u prostoru.

1. **Žiroskop**

   Žiroskop je senzor koji se uglavnom koristi za merenje ugaone brzine, tj. brzine rotacije oko osa senzora kao i za navigaciju. Jedinica mere je **rad/s.** Ovi uređaji mogu da mere ugaonu brzinu duž jedne, dve ili tri koordinatne ose. Žiroskopi su ranije bili mehaničko-inercijalni, a danas su to elektronski i optički uređaji. Osnovni tipovi žiroskopa su:

   - Rotacioni (klasični) žiroskop
   - Vibrirajući žiroskop
   - Optički žiroskop

   Važnije osobine žiroskopa:

   - Dvoosni žiroskopi sadrže dva, a troosni tri jednoosna žiroskopa. Svaki od njih je postavljen normalno u odnosu na druge. Rotacioni i optički žiroskop su uglavnom troosni.
   - Merni opseg je najveća ugaona brzina koju žiroskop može da izmeri (u rad/s). Merni opseg za ljudske pokrete je obično interval $(-1, 1)$. Različiti uređaji mogu imati različit znak smera rotacije.
   - Maksimalan broj očitavanja signala sa žiroskopa u jedinici vremena zove se frekvencija očitavanja.
   - Temperaturni opseg predstavlja podatak koji pokazuje kolika je minimalna i maksimalna temperatura pri kojima žiroskop može da radi. Interval čiji su krajevi minimalna i maksimalna temperatura zove se temperaturni opseg.

2. **Akcelerometar**

   Akcelerometar je senzor kojim se meri ubrzanje (akceleracija) objekta, odnosno ljudskog tela u pokretu. Njegov rad uglavnom se bazira na merenju inercijalnih sila, ali postoje i optički akcelerometri, kao i oni koji su u obliku integrisanog kola. Opseg kod praćenja ljudskih aktivnosti je obično interval (-1,1). Jedinica mere je g.

3. **Magnetometar**

   Magnetometar je senzor pomoću koga se određuje orijentacija u odnosu na geomagnetno polje Zemlje. Pomoću ovog uređaja se dobijaju komponente vektora magnetnog polja na mestu na kome se uređaj nalazi u datom vremenskom trenutku. Senzor registruje uglove koje obrazuje vektor položaja senzora sa koordinatnim osama navigacionog sistema. Opseg je interval $(-1, 1)$. Jedinica mere je stepen.

3 Klaster analiza

Postoje razni skupovi objekata (entiteta) u kojima se, prema nekoj osobini, mogu formirati podskupovi (grupe objekata). Veoma često postoji potreba da se identifikuju ti podskupovi (grupe). Najčešće je potrebno podeliti dati skup objekata na homogene i dobro razdvojene celine koje zovemo klasteri, tako da su objekti koji se nalaze u istom klasteru, prema nekoj osobini slični, a objekti koji se nalaze u različitim klasterima različiti.

Postoji više metoda za određivanje klastera. Najbolje rezultate daje metoda koja se zove klaster analiza ili analiza grupisanja. Postupak određivanja klastera zovemo klasterovanje. Postoje razni algoritmi (metode) klasterovanja. Primenom bilo koje metode klasterovanja nastoji se da se postigne što veća homogenost klastera i različitost između klastera. Izbor optimalnog algoritma klasterovanja je veoma složen zadatak, jer optimalnost nije jednoznačna i u velikoj meri zavisi od datog skupa objekata i njihovih bitnih osobina.

Algoritmi klasterovanja su međusobno dosta različiti i veoma je teško formirati jedan kriterijum po kome bi bilo moguće odabrati optimalan algoritam za konkretnu primenu. Ako ne postoje jasni kriterijumi, izbor optimalnog algoritma je složen proces sa neizvesnim ishodom.

Klaster analiza ima mnogobrojne primene u različitim oblastima, kao što su: prepoznavanje ljudskih aktivnosti, prepoznavanje oblika, istraživanje podataka, mašinsko učenje, statistika, ekonomija, biologija, medicina, psihologija, sociologija i drugo.

U procesu klaster analize potrebno je rešiti dva osnovna problema:

- Izbor odgovarajuće mere udaljenosti (sličnosti),
- Izbor algoritma klasterovanja.

Klaster analizom se pronalazi povezanost objekata bez utvrđivanja uzroka te povezanosti. Najčešće, broj klastera i zajedničke osobine objekata u klasteru nisu poznati pre početka klasterovanja.

Navodimo opšte osobine klaster analize:

- Karakteristike skupa objekata - osobine objekata koje su značajne prilikom izbora metoda klasterovanja
- Spoljašnje karakteristike metode klasterovanja - one koje su bitne korisniku koji je više zainteresovan za izbor metode koja njemu odgovara, a manje za detalje rada algoritma klasterovanja
- Unutrašnje karakteristike metode klasterovanja - one koje su važne istraživačima metodologije klaster analize

Važnije posebne osobine klaster analize:

- Osnovni cilj je da se potvrdi da je klasterovanje dobro poslužilo željenoj svrsi.
- Očekivani rezultat klaster analize je dobijanje jedne optimalne particije.
- Od broja objekata u velikoj meri zavisi izbor algoritma klasterovanja.
- Moguća su razvrstavanja na osnovu podataka o pojedinačnim objektima i na osnovu podataka o bliskosti objekata.
- Dati skup objekata može biti kompletan ili nekompletan skup objekata koji se klasteruje.
- Ukoliko postoje značajne razlike u broju objekata u pojedinim klasterima, takvi skupovi su složeniji za analizu.
- Posebno je složeno klasterovanje ukoliko postoji preklapanje klastera.
- Svaki postupak treba da zadovoljava kriterijume prihvatljivosti. Korisnik određuje koji su kriterijumi prihvatljivosti važni za njegov skup objekata.

Poželjne karakteristike algoritma klasterovanja:

- Mogućnost primene u radu sa različitim skupovima objekata,
- Relativno mali broj ulaznih parametara,

- Jednostavnost korišćenja i interpretacije rezultata.

Ako je mali skup podataka (do 100 objekata), veliki broj algoritama klasterovanja daje dobre rezultate. Međutim to nije slučaj ako baze sadrže više miliona podataka.

Kod većine algoritama klasterovanja polazi se od pretpostavke da su poznate vrednosti određenih parametara. Najčešće je to broj potrebnih klastera. Neretko, rezultati klasterovanja su u velikoj meri zavisni od vrednosti ulaznih parametara. Ulazne parametre često nije lako odrediti. To predstavlja opterećenje za istraživački postupak i za posledicu ima da rezultat klasterovanja nije zadovoljavajući. U realnim bazama podataka mnogi objekti nisu sadržani ni u jednom klasteru ili obrazuju izuzetno malobrojne klastere. Pojedini algoritmi klasterovanja u tom slučaju daju nezadovoljavajuće rezultate klasterovanja.

Algoritme klasterovanja u kojima algoritam mora sam proceniti broj klastera zovemo nenadgledani. Međutim, mnogi algoritmi klasterovanja koji su trenutno u upotrebi zahtevaju da dodatne informacije dostavi korisnik. Ovo bi bilo razumno kada bi korisnik mogao da vizuelno istraži podatke i pruži razumnu pretpostavku o broju klastera, međutim, u problemima sa velikim brojem podataka to nije izvodljivo. Zahtev od korisnika da unese vrednosti bilo kog parametra je takođe loša, ali možda neizbežna praksa. Bilo koju metodu koja zahteva dodatne podatke od korisnika, osim podataka koje analiziramo, zovemo nadgledana.

Ovaj tip analize podseća na metode klasifikacije objekata. Ipak, ove dve metode su dosta različite jedna od druge. Prvo, u klasifikaciji, poznato nam je, još na početku, u koliko klasa ili grupa treba klasifikovati objekte i koje objekte gde razvrstati. U klaster analizi, broj klastera je uglavnom nepoznat, kao i koji element gde treba razvrstati. Drugo, u klasifikaciji, cilj je da se klasifikuju novi objekti u jednu od datih klasa na osnovu prethodnog iskustva. Prilikom klasterovanja, obično, ne postoje pouzdane informacije o strukturi skupa objekata koji klasterujemo.

3.1 Osnovni principi klasterovanja

Definišimo prvo pojam rasplinutog skupa, koji ćemo koristiti u ovoj sekciji. Neka je A neprazan skup. Svako preslikavanje

$$\bar{A} : A \longrightarrow [0, 1] \tag{1}$$

zovemo **fazi (fuzzy)**, odnosno **rasplinuti podskup** od A.

Skup A je univerzum za $\bar{A}$. Za $a \in A$, $\bar{A}(a)$ je stepen pripadanja elementa a rasplinutom skupu $\bar{A}$. Rasplinuti skup je identifikovan sa preslikavanjem - uopštenjem karakteristične funkcije. Ako je $\bar{A}(A) = \{0, 1\}$, rasplinuti skup je upravo jedna karakteristična funkcija.

Pretpostavimo da je

$$A = \{a_1, a_2, ..., a_n\} \tag{2}$$

konačan skup objekata i $A_i, i = 1, 2, ..., k$, klasteri skupa A. Tada su ispunjeni sledeći uslovi:

$$1. \quad A_i \neq \emptyset, \tag{3}$$

$$2. \quad A = \bigcup_{i=1}^{k} A_i. \tag{4}$$

Ako je ispunjen uslov

$$3. \quad A_j \cap A_l = \emptyset, \ j, l = 1, 2, ..., k; \ j \neq l, \tag{5}$$

onda kažemo da je to tvrdo klasterovanje.

Ako objekat a_i pripada klasteru A_j, kažemo da je stepen pripadanja v_{ji} objekta a_i klasteru A_j jednak 1, tj. $v_{ji} = 1$. U suprotnom je $v_{ji} = 0$. To se može predstaviti pomoću matrice

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & & \vdots \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{bmatrix}. \tag{6}$$

Pri tome su ispunjeni sledeći uslovi

$$v_{ji} \in \{0,1\},\ 1 \leqslant j \leqslant k,\ 1 \leqslant i \leqslant n, \tag{7}$$

$$\sum_{j=1}^{k} v_{ji} = 1,\ 1 \leqslant i \leqslant n, \tag{8}$$

$$\sum_{i=1}^{n} v_{ji} > 0,\ 1 \leqslant j \leqslant k. \tag{9}$$

Iz prethodnog se može videti da svaki objekat mora pripadati samo jednom klasteru, što je označeno sa $v_{ij} = 1$, dok se nepripadnost ostalim klasterima označava sa nulom. Takođe, svaki klaster mora sadržati bar jedan objekat, odnosno ne sme postojati prazan klaster. Za razliku od čvrstog klasterovanja, rasplinuto ili fazi (eng. *fuzzy*) klasterovanje dozvoljava da jedan objekat istovremeno pripada većem broju klastera sa odgovarajućim stepenom pripadanja iz intervala $[0, 1]$, odnosno važi:

$$v_{ji} \in [0,1],\ 1 \leqslant j \leqslant k,\ 1 \leqslant i \leqslant n. \tag{10}$$

Matrica V kod rasplinutog klasterovanja takođe zadovoljava uslove (8) i (9), i takođe ne postoji prazan klaster.

3.2 Mere rastojanja i sličnosti

Grupisanje elemenata proučavanog skupa u klastere se vrši prema sličnosti ili rastojanju (distanci) između različitih objekta, a zovu se još i mere bliskosti. Nije formiran jedinstven stav oko toga koja je mera sličnosti najadekvatnija u postupku klasterovanja. Sličnost između objekata je obrnuto proporcionalna distanci između njih. Što su objekti sličniji, distanca između njih je manja. Da bi se formirali klasteri, potrebno je odabrati odgovarajuću meru sličnosti ili udaljenosti. Zato je potrebno poznavanje predmeta istraživanja. Odabir mere sličnosti između dva elementa je veoma važan za klaster analizu, zato što često različite metode daju i različite rezultate.

Neka je A skup podataka. Mera rastojanja se može opisati realnom funkcijom

$$d : A \times A \to R, \tag{11}$$

koja zadovoljava sledeće uslove:

(1) $(\forall x, y \in A)\ d(x, y) \geqslant 0$,

(2) $(\forall x \in A)\ d(x, x) = 0$,

(3) $(\forall x, y \in A)\ d(x, y) = d(y, x)$.

Ako se uslov (2) zameni uslovom (2') i doda uslov (4)

(2') $(\forall x, y \in A)\ d(x, y) = 0 \Longleftrightarrow x = y$,

(4) $(\forall x, y, z \in A)\ d(x, y) \leqslant d(x, z) + d(z, y)$,

tada meru rastojanja nazivamo metrikom. Metrika jeste mera rastojanja, ali mera rastojanja ne mora biti metrika. Pored mere rastojanja definiše se i mera sličnosti. Za meru

$$s : A \times A \to [0, 1] \tag{12}$$

kažemo da predstavlja meru sličnosti ako su ispunjeni uslovi:

1. $(\forall x, y \in A)\ 0 \leqslant s(x, y) \leqslant 1$,

2. $(\forall x, y \in A)\ s(x, y) = 1 \Longleftrightarrow x = y$

3. $(\forall x, y \in A)\ s(x, y) = s(y, x)$.

U nastavku su opisane neke od najčešće korišćenih mera rastojanja koje se koriste prilikom klasterovanja.

Neka su

$$x = (x_1, x_2, ..., x_n) \text{i} y = (y_1, y_2, ..., y_n) \tag{13}$$

proizvoljni elementi skupa podataka A.

1. **Euklidova metrika**:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2} \tag{14}$$

2. **Apsolutna metrika**:

$$d(x, y) = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n|. \tag{15}$$

3. **Metrika Minkovskog**:

   Pve dve metrike su specijalni slučajevi Metrike Minkovskog:

$$d(x, y) = \Big(\sum_{s=1}^{n} (x_s - y_s)^r \Big)^{\frac{1}{r}}. \tag{16}$$

4. **Čebiševljeva ili maksimalna udaljenost**:

   Definisana je kao maksimalna udaljenost dva objekta x i y. Računa se pomoću obrasca:

$$d(x, y) = \max_{1 \leqslant i \leqslant n} |x_i - y_i|. \tag{17}$$

3.3 Tipovi klastera

Tipovi klastera se razlikuju po principima koji se koriste prilikom definisanja klastera. Navodimo osnovne tipove klastera:

- **Dobro razdvojeni klasteri:**

  Klaster je skup objekata za koje važi da su na manjoj udaljenosti od ostalih objekata u klasteru nego od objekata koji nisu u klasteru.

- **Klasteri bazirani na centru:**

  Klaster je skup objekata za koje važi da su manje udaljeni od centra klastera u odnosu na centre ostalih klastera. Objekti jednog klastera su bliži centru svog klastera nego centrima drugih klastera, ali mogu biti bliži elementima drugog klastera nego svom centru.

- **Klasteri bazirani na grafovima (zasnovani na susedstvu):**

  Klaster je skup objekata za koje važi da su bliži jednom ili nekoliko objekata u svom klasteru nego bilo kom objektu koji nije u klasteru kome on pripada.

- **Klasteri opisani funkcijom cilja:**

  Opisivanje klastera funkcijom cilja predstavlja implicitan metod, tj. potrebno je pronaći klastere koji minimizuju (maksimizuju) funkciju cilja. Dva različita klasterovanja mogu dati dve različite vrednosti funkcije cilja, a zatim da odaberemo onaj način koji daje najbolju vrednost funkcije cilja.

- **Klasteri bazirani na gustini:**

  Klasteri su oblasti sa velikom gustinom objekata koje su razdvojene oblastima sa malom gustinom objekata. Koriste se kada su klasteri nepravilni i imaju objekte van granica.

- **Konceptualni klasteri:**

  Klasterovanje na osnovu zajedničkih osobina, kada je klaster definisan nekim svojstvom. Objekti imaju neko zajedničko svojstvo koje određuje klaster. Potrebno je odrediti klastere koji imaju neku zajedničku osobinu.

3.4 Klasifikacija algoritama za klasterovanje

U literaturi se mogu naći mnogobrojni i raznovrsni algoritmi koji služe za klasterovanje. Međutim, ne postoji najbolji algoritam za klasterovanje, jer performanse algoritma variraju na različitim skupovima podataka, zato što klasterovanje zavisi od više faktora kao što su struktura, vrsta podataka ili dimenzionalnost.

Postoje razni kriterijumi za klasifikaciju algoritama klasterovanja. U zavisnosti od odabranog kriterijuma moguće su razne klasifikacije. Neki od kriterijuma bi bili: tipovi podataka, tipovi promenljivih, sličnost između podataka, preklapanje klastera. Npr. ako je kriterijum preklapanje klastera, razlikujemo fazi (rasplinuto, fuzzy) i tvrdo klasterovanje. Kod tvrdog klasterovanja jedan element može pripadati samo jednom klasteru, a kod rasplinutog jedan element može pripadati više klastera, naravno sa različitim stepenom pripadanja. Do sada većina algoritama klasterovanja pripada tvrdom klasterovanju.

U literaturi je dosta prisutna sledeća klasifikacija metoda klasterovanja:

- Hijerarhijske metode,
- Nehijerarhijske metode,
- Metode zasnovane na mreži,
- Metode zasnovane na gustini,
- Metode zasnovane na modelu.

Najčešće se jednostavno razlikuju samo hijerarhijske i nehijerarhijske metode.

3.5 Hijerarhijske metode

Osnovna ideja hijerarhijskih metoda klasterovanja je da se dobije hijerarhijski odnos između (grupa) podataka u cilju što prirodnije klasterizacije. Prvo se računaju sve distance između objekata, a onda se postupcima udruživanja ili razbijanja obrazuju klasteri. Ovi algoritmi se dele na aglomerativne (metode spajanja ili udruživanja) i hijerarhijske metode razdvajanja.

Aglomerativno klasterovanje započinje jednostrukim klasterima gde je svaki objekat poseban klaster. Prvi korak je grupisanje dva klastera sa najvećom merom sličnosti (najmanjom distancom) između klastera. Ukoliko najveći koeficijent sličnosti ima nekoliko parova klastera, na slučajan način se bira par klastera koji će prvo biti spojeni. Zatim se ponovo određuje koeficijent sličnosti (rastojanja) između dobijenih klastera. Svaka metoda ima specifičan postupak za određivanje sličnosti između klastera. Postupak se ponavlja do postizanja kriterijuma zaustavljanja. Podeljeno klasterisanje započinje jednim klasterom svih podataka nakon čega sledi rekurzivna deoba klastera tako da podaci pripadnu najbližem klasteru. Proces se nastavlja sve dok se ne postigne kriterijum zaustavljanja (često, traženi broj klastera k).

Rezultat hijerarhijskih metoda je dendrogram (stablo klastera), koji predstavlja ugnježđeno grupisanje objekata po nivoima sličnosti na kojima se grupisanje menja. Grupisanje objekata u jednom nivou podataka dobija se sečenjem dendrograma na željeni nivo sličnosti.

Prednosti hijerarhijskog grupisanja:

- Ugrađena fleksibilnost u pogledu nivoa klasterizacije,
- Lako izračunavanje rastojanja (distanca, sličnost) podataka,
- Primenljivost na bilo koju vrstu atributa.

Nedostaci hijerarhijskog grupisanja:

- Nejasnoća kriterijuma prekida metode,
- Ako su objekti jednom spojeni, ne mogu se razdvajati, kao i obrnuto, razdvojeni objekti ne mogu se spajati.

Kako se aglomerativne metode razlikuju uglavnom prema meri rastojanja između klastera, navodimo samo kako se ona određuje za svaku od navedenih metoda. Najčešće primenjivane hijerarhijske aglomerativne metode su sledeće:

- **Metoda jednostrukog (prostog) povezivanja, metoda minimuma ili metoda najbližeg suseda** (engl. *Single linkage*):

  Mera rastojanja između dva klastera predstavlja najkraće rastojanje između parova objekata koji pripadaju ovim klasterima. To je zapravo udaljenost između najsličnijih elemenata.

$$d(C_i \cup C_j, C_k) = min(d(C_i, C_k), d(C_j, C_k)) \tag{18}$$

- **Metoda potpunog (kompletnog) povezivanja ili metoda maksimuma** (engl. *Complete linkage*):

  Mera rastojanja između dva klastera je najveća distanca između parova objekata, od kojih jedan pripada jednom, a drugi drugom posmatranom klasteru.

$$d(C_i \cup C_j, C_k) = max(d(C_i, C_k), d(C_j, C_k)) \tag{19}$$

- **Metoda prosečnog povezivanja ili Metoda proseka** (engl. *Group average*):

  Rastojanje se određuje kao prosečno rastojanje svih objekata jednog klastera od svih objekata drugog klastera.

$$d(C_i \cup C_j, C_k) = \frac{1}{(n_i + n_j)n_k} \sum_{u \in C_i \cup C_j} \sum_{v \in C_i \cup C_j} d(u, v) \tag{20}$$

- **Metoda centroida** (engl. *Gower method*):

  Primenom ove metode se dva klastera spajaju u novi klaster ako su njihovi centroidi najmanje udaljeni međusobno u odnosu na međusobnu udaljenost svih mogućih parova klastera koji postoje na posmatranom nivou udruživanja.

$$d(C_i \cup C_j, C_k) = d^2(t_{ij}, t_k), \tag{21}$$

  gde je t_{ij} centar klastera $C_i \cup C_j$, a t_k centar C_k. Udaljenost između dva klastera jednaka je kvadratu udaljenosti između centara klastera.

- **Metoda Ward-a ili Metoda minimalne varijanse**:

  Polazna osnova su svi mogući jednočlani klasteri. Pri tome se ne razmatra distanca između klastera, nego se maksimalno povećava homogenost klastera. Primenom ove metode klasteri se spajaju u slučaju kada tim spajanjem dolazi do minimalnog rasta zbira kvadrata greške (SSE) u klasteru, pri čemu je:

$$SSE = \sum_{i=1}^{n} \sum_{j=1}^{m_i} (x_{ij} - \bar{x}_i)^2, \tag{22}$$

  pri čemu je x_{ij} j-ti objekat u i-tom klasteru, n je broj klastera, $\bar{x}_i$ je centar i-tog klastera, a m_i je broj objekata u i-tom klasteru.

- **Algoritam klasterovanja za segmentaciju slika regionom**:

  Segmentacija slika pomoću regiona čini skup metoda pomoću kojih se izdvajaju oblasti slike koje su homogene prema nekim karakteristikama. Ovu grupu čine: segmentacija pomoću rasta regiona i segmentacija pomoću razdvajanja i spajanja regiona. Metodom segmentacije rasta regiona se vrši

grupisanje susednih piksela sličnih osvetljenosti (boja), na osnovu toga se formiraju regioni. Grupisanje počinje spajanjem po dva piksela istih karakteristika i tako nastaje atomski region. Zatim se posmatraju dva susedna regiona R_1 i R_2. Broj ivičnih piksela tih regiona, u oznaci B_1 i B_2, zovemo obimi tih regiona. Sa C označimo dužinu zajedničke granice regiona, a sa D dužinu zajedničke granice gde je razlika između karakteristika piksela sa obe strane granice manja od unapred date vrednosti. Ako je ispunjen uslov:

$$\frac{D}{min(P_1, P_2)} > \varepsilon_2, \tag{23}$$

regioni R_1 i R_2 se spajaju. Ovde je ε_2 konstanta čija je vrednost unapred data. Zatim se ispituju ostali atomski regioni, nakon čega se ispituju regioni većih dimenzija sve dok je spajanje moguće. Na ovaj način se manji regioni priključuju većim. Spajanje regiona sličnih dimenzija nije dozvoljeno. Da bi se mogli spojiti dva regiona sličnih veličina koji su razdvojeni tzv. slabom granicom, mora biti ispunjen uslov:

$$\frac{D}{C} > \varepsilon_3, \tag{24}$$

gde je ε_3 konstanta čija je vrednost unapred data.

3.6 Nehijerarhijske metode

Kod ovih metoda pretpostavka je da je poznat broj klastera. Za razliku od prethodno navedenih metoda, nehijerarhijske metode omogućavaju da objekti mogu prelaziti iz jednog kandidata za klaster u drugi, ako to neki kriterijum zahteva.

Prilikom primene nehijerahijske metode klasterovanja, prvo se uradi početna podela na unapred određen broj klastera. Postoje dva načina za razvrstavanje objekata u inicijalnoj podeli. Prvi je da se privremeno, na slučajan način, odrede objekti koji predstavljaju tačke grupisanja. Određuje se onoliko tačaka grupisanja koliko je unapred definisano klastera. Drugi način razvrstavanja u inicijalnoj podeli je da se razvrstavanje odvija na osnovu nekog unapred zadatog kriterijuma. Jedna od mogućih strategija za određivanje tačaka grupisanja jeste da

na početku koristimo hijerarhijski pristup kako bismo utvrdili koliko klastera postoji, a zatim da koristimo centre dobijene iz ovog kao početne centre u nehijerarhijskoj metodi.

Da bi se optimizovala funkcija cilja, objekti se mogu premeštati iz jednog klastera u drugi. Zatim se određuje distanca između svakog objekta i svakog klastera. Ukoliko je objekat najbliži određenom klasteru, biće smešten u taj klaster. Posle premeštanja objekta iz jednog klastera u drugi, određuju se centroidi klastera iz kog je objekat premešten i klastera u koji je premešten objekat. Potom se određuje rastojanje od centroida klastera i preraspodeljuju objekti sve dok je to potrebno po posmatranom kriterijumu.

Za razliku od hijerarhijskih metoda, kod ovih metoda nije neophodno grafičko prikazivanje podataka pomoću stabla. Nehijerarhijske metode klasterovanja su brže i pouzdanije od hijerarhijskih metoda. Glavni problem sa kojim se suočavaju sve nehijerarhijske metode klasterovanja je kako odrediti broj klastera.

Najviše primenjivane nehijerarhijske metode su metoda k-sredina (eng. *k-means method*), u kojoj prosečna vrednost objekata u klasteru je predstavnik klastera i metoda *k-medoida*, u kojoj centar klastera je neki objekat klastera. Većina nehijerarhijskih metoda klasterovanja korisati neku od ove dve metode ili pak modifikuje neku od njih. Ovi algoritmi dobro funkcionišu ukoliko skupovi podataka nisu mnogo veliki. Prilikom klasterovanja velikog skupa podataka, potrebno je proširenje ovih metoda. Takođe, često korišćene metode su *Spektralna metoda klasterovanja* i *Metoda fazi k-sredina*.

- **Metoda k-sredina** (engl. *k-means*):

  Sa k je označen broj klastera koji želimo da nađemo. Postupa se na sledeći način:

  1. Nasumice izaberemo k instanci iz datih podataka i one predstavljaju polazne centroide.
  2. Zatim preraspodeljujemo svaku od n instanci najbližim centroidima i tako dobijamo klastere.

3. U sledećem koraku izračunavamo centroide tako dobijenih klastera. Ti centroidi ne moraju odgovarati polaznim.
4. Potom, ponovo preraspodeljujemo instance najbližim centroidima i tako dobijamo nove klastere.
5. Dalje određujemo nove centroide unutar klastera.
6. Poslednja dva koraka se smenjuju sve dok se sredine menjaju, tj. sve dok nijedna instanca ne promeni klaster.

U algoritmu k-sredina najbliže rastojanje se određuje kao Euklidsko rastojanje, kosinusno rastojanje, korelacija, itd. Najčešće se koristi Euklidsko rastojanje. Rešenje dobijeno ovom metodom (određivanje klastera, odnosno podela objekata u klastere) ne mora biti jedinstveno. Algoritam će samo pronaći lokalni minimum sume kvadrata grešaka (SSE). Ukoliko nije poznat broj klastera, preporučuje se da primenimo metodu koristeći različit broj klastera kao i različite tačke grupisanja, a zatim od njih izabrati onaj koji daje minimalnu vrednost kvadrata greške. Ovom metodom se optimizuje prosečna udaljenost objekata u istom klasteru

$$\sum_{s=1}^{k} \frac{1}{m_s} \sum_{i=1}^{m_s} \sum_{j=1}^{m_s} \|x_{si} - x_{sj}\|^2. \tag{25}$$

Cilj je da se smanji greška (SSE), odnosno zbir kvadrata rastojanja svake instance od centra svog klastera. Greška se određuje koristeći sledeći obrazac:

$$SSE = \sum_{s=1}^{k} \sum_{i=1}^{m_s} \|x_{si} - \mu_s\|^2. \tag{26}$$

- **Metoda k-medoida** (engl. *k-medoids*):

  Ovaj algoritam je veoma sličan algoritmu k-sredina. Uglavnom se razlikuju po načinu izbora predstavnika klastera. Svaki klaster predstavljen je najcentriranijim objektom u klasteru, a ne srednjom vrednošću koja, možda, ne pripada klasteru. Ova metoda zahteva od korisnika da navede broj klastera. Medoid je objekat koji je izabran iz skupa podataka da

predstavlja klaster. Algoritam bira k medoida da predstavljaju klastere. Zatim se formiraju klasteri dodeljivanjem svakog od preostalih objekata najbližem medoidu. Potom se određuju novi medoidi kao najcentriraniji objekti u formiranim klasterima. Poslednja dva koraka se smenjuju sve dok se medoidi menjaju.

- **Spektralna metoda klasterovanja:**

  Ovaj algoritam spada u klasu spektralnih algoritama zato što je metoda bazirana na analizi spektra Laplasove matrice grafa.

  Definišimo prvo osnovne pojmove koji se koriste u spektralnom klasterovanju grafova. Neka je $G = (V, E)$ težinski graf sa skupom čvorova $V = \{v_1, v_2, ..., v_n\}$ i matricom susedstva W. W je simetrična matrica. Sa w_{ij} je označena težina grane (v_i, v_j). Ako je $w_{ij} = 0$, nema grane između čvorova v_i i v_j .

  Sopstvene vrednosti matrice W jesu brojevi takvi da za jednačinu $Wx = \lambda x$ postoji rešenje koje nije nula za vektor x, i u tom slučaju rešenje x je odgovarajući sopstveni vektor. Grafu G odgovara njegova matrica susedstva W. Za sopstvene vrednosti ove matrice kažemo da su sopstvene vrednosti grafa G. Tada je W simetrična matrica, ima realne sopstvene vrednosti i sopstvene vektore. Jednakost $Wx = \lambda x$ možemo predstaviti na sledeći način:

$$\lambda x_i = \sum_{(i,j)\in E(G)} x_j,\ i \in V(G). \tag{27}$$

  Sopstvene vrednosti su nule karakterističnog polinoma

$$p(G, \lambda) = det(\lambda I - W) = \prod(\lambda - \lambda_i). \tag{28}$$

  Skup sopstvenih vrednosti, uključujući i njihove višestrukosti čine spektar grafa G. Ako je višestrukost sopstvene vrednosti jednaka jedan, kažemo da je sopstvena vrednost prosta.

  Dijagonalna matrica D, takva da je

$$d_{ii} = \sum_{j=1}^{n} w_{ij}, \tag{29}$$

zove se matrica stepena, a d_{ii} stepen čvora v_i.

Laplasovu matricu definišemo na sledeći način: $L = D - W$. Takođe definišemo i normalizovanu Laplasovu matricu:

$$L = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}} \tag{30}$$

Kako algoritam spektralnog klasterovanja zahteva graf, prvo se od datog skupa podataka formira graf. Taj graf se zove graf sličnosti.

Pored grafa smatramo da je poznat i broj klastera. Cilj ovog algoritma je grupisanje čvorova grafa tako da važi :

- Zbir težina grana čiji krajnji čvorovi pripadaju istom klasteru treba da bude što veći.
- Zbir težina grana koje povezuju čvorove koji pripadaju različitim klasterima treba da bude što manji.

Navodimo primer algoritma spektralnog klasterovanja:

[Ng et al., 2002]

Input: W i k;

for all i, j **do**

if $i = j$ **then**

$D_{ii} := \sum_l w_{il}$;

else

$D_{ij} := 0$;

end if

end for

$L = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}$

Odrediti matricu H čije su kolone prvih k sopstvenih vektora matrice L;

Normalizovati vrste matrice H;

Klasterovati vrste matrice H primenom algoritma k-srednjih vrednosti 1;

Output: Klasteri $C_1, C_2, ..., C_k$.

- **Metoda fazi k-sredina** (engl. *Fuzzy k-means*):

  Algoritam fazi k-sredina (FKS) pripada grupi rasplinutih klasterovanja. FKS se razlikuje od većine drugih algoritama zato što raspoređuje objekte u nekoliko klastera, dodeljujući im stepen pripadnosti klasteru $v_{ji} \in [0, 1]$, gde je v_{ji} stepen pripadanja objekta a_i klasteru A_j, $i = 1, ..., n$; $j = 1, ..., k$. Pri tome važe relacije (8) i (9) koje obezbeđuju da zbir stepena pripadnosti nekog objekta svim klasterima mora biti jedan. Takođe nijedan klaster ne može da bude prazan.

  FKS predstavlja iterativan algoritam i sastoji se od sledećih koraka:

  1. Na slučajan način formiramo inicijalnu matricu

     $$V = [v_{ji}], 1 \leqslant j \leqslant k,\ 1 \leqslant i \leqslant n, \tag{31}$$

     gde je k unapred dat broj klastera i n broj objekata u skupu $A = \{a_1, a_2, ..., a_n\}$.

  2. Izračunamo centroide klastera.

  3. Izračunamo stepene pripadnosti v_{ji}.

  4. Uporedimo $V_{(t+1)}$ i $V_{(t)}$, gde t predstavlja redni broj iteracije.

  5. Ukoliko važi

     $$\|V_{(t+1)}) - V_{(t)}\| < \varepsilon \tag{32}$$

     ili je dostignut maksimalni broj iteracija, završavamo algoritam. U suprotnom, sledi povratak na korak 2.

  Vrednost ε se utvrđuje pre izvođenja algoritma.

3.7 Metode zasnovane na mreži

Glavna karakteristika ovih metoda je deljenje prostora podataka u konačan broj ćelija koje formiraju mrežu. Sve operacije klasterovanja se zatim izvode na ćelijama ove mreže. Nakon što izvršimo deljenje prostora u mrežu, potrebno je da objekte dodelimo odgovarajućim ćelijama i zatim računamo gustinu u svakoj ćeliji. Dalje uklanjamo sve ćelije koje imaju manju gustinu od tražene. Na kraju formiramo klastere od susednih grupa gustih ćelija, obično tako što minimiziramo određenu ciljnu funkciju.

Ključna prednost ove metode je kratko vreme izvođenja. To vreme ne zavisi od broja objekata, već samo od broja ćelija.

Karakterističan primer ove metode je STING (Statistical Information Grid).

3.8 Metode zasnovane na gustini

Ako su podaci predstavljeni u prostoru, klasteri su definisani kao guste oblasti u tom prostoru. Gustina se ovde odnosi na fizički pojam gustine, a ne na određenu statističku raspodelu. Ovi algoritmi mogu otkriti grupe podataka proizvoljnog oblika i guste oblasti (klasteri) će biti razdvojeni međusobno oblastima sa manjom gustinom. Realizacija ove ideje za podelu konačnog skupa tačaka zahteva koncepte gustine, povezanosti i granice. Oni su usko povezani sa najbližim susedima (podacima) datog podatka. Zato algoritmi, zasnovani na gustini, mogu da otkriju podskupove podataka proizvoljnih oblika.

Pored dobrih osobina, ove metode imaju i neke nedostatke. Jedan gusti klaster koji se sastoji od dve susedne oblasti sa znatno različitim gustinama, a obe su veće od praga, ne daje dovoljno korisnih informacija.

3.9 Metode koje su zasnovane na modelu

Ovim metodama se svim klasterima pridružuju odgovarajući modeli i pronalazi najbolje fitovanje podataka za svaki model. Prilikom primene ovakvih metoda potrebno je imati predstavu o strukturi podataka. Algoritmi klasterovanja koji se

zasnivaju na modelu mogu odabrati klastere koristeći funkciju gustine koja predstavlja raspodelu podataka u prostoru. Ovi modeli zasnivaju se na zajedničkoj gustini raspodele podataka ili neuronskim mrežama. Upotreba neuronskih mreža u početku je bila motivisana apstraktnim pokušajem da se modelira način na koji mozak grupiše objekte.

4 Klasterovanje bazirano na grafovima

U ovoj sekciji razmatramo klasterovanje potpunog težinskog grafa. Prvo predstavljamo nov algoritam klasterovanja za slučaj kada nije poznat broj klastera, a potom i u slučaju kada je broj klastera unapred dat.

4.1 Grafovi

Uzmimo da je $V \neq \emptyset$ i E binarna relacija na skupu V. Kažemo da je **graf** uređen par $G = (V, E)$. Elementi skupa V se zovu **čvorovi** grafa, a elementi skupa E **grane** grafa. Čvorove grafa $v_1, \ldots, v_n$ možemo predstaviti tačkama u prostoru ili ravni. Ukoliko $(v_i, v_j) \in E$, neprekidnom linijom spajamo tačke koje predstavljaju čvorove v_i i v_j. Ova linija se orijentiše strelicom od v_i ka v_j i ona ne prolazi kroz neki treći čvor grafa. Ako $(v_i, v_j) \notin E$, čvorovi v_i i v_j nisu direktno povezani.

Ako paru čvorova v_i, v_j odgovaraju dve grane (v_i, v_j) i (v_j, v_i), ponekad se ne povlače dve linije između čvorova v_i i v_j, već se jedna linija između njih orijentiše dvostruko ili se ostavlja bez orijentacije. Grana, koja spaja čvor sa samim sobom, naziva se **petlja.**

Graf $G = (V, E)$ je **neusmeren** ili neorijentisan ukoliko je E simetrična relacija. Graf $G = (V, E)$ je **usmeren** ili orijentisan ako je E antisimetrična relacija. Petlju nije neophodno orijentisati pošto za bilo koju orijentaciju, petlja počinje i završava se u istom čvoru. Zbog toga se prilikom grafičkog predstavljanja petlje strelica izostavlja. Postoje, naravno, i grafovi koji nisu ni neusmereni ni usmereni. Uglavnom je praksa da se umesto grana suprotnih orijentacija stavljaju neorijentisane grane.

U zavisnosti od toga da li je skup čvorova V beskonačan ili konačan i grafovi mogu biti beskonačni ili konačni. Za dva čvora neusmerenog grafa bez petlji kažemo da su **susedni** ako između njih postoji grana. Čvorovi su susedni ukoliko su krajevi neke grane. Kažemo da se grana stiče u nekom čvoru ako je taj čvor krajnja tačka te grane. Tada, takođe, kažemo da su grana i čvor susedni. Broj

susednih čvorova za čvor v zove se **stepen čvora** v i označavamo ga sa $d(v)$. Stepen čvora je jednak broju grana koje se stiču u tom čvoru. Dve **grane su susedne** ukoliko imaju zajednički čvor. Ukoliko u usmerenom grafu grana u spaja čvorove v_i i v_j i orijentisana je od v_i ka v_j, kažemo da grana u izlazi iz čvora v_i, a ulazi u čvor v_j. Takođe se kaže da je v_i početni, a v_j završni čvor grane u. Broj grana koje ulaze u neki čvor zovemo **ulazni stepen čvora**. Broj grana koje izlaze iz nekog čvora zovemo **izlazni stepen** čvora. Obično smatramo da je petlja ujedno i ulazna i izlazna grana za svoj čvor. U konačnom neusmerenom grafu bez petlji broj čvorova neparnog stepena mora biti paran. Navodimo još jedan način predstavljanja grafa $G = (V, E)$ pomoću **matrice povezanosti** $A_{G_{|V|\times|V|}}$ (kraće A_G), gde su elementi matrice dati na sledeći način:

$$A_G[i,j] = \begin{cases} 1, & (v_i, v_j) \in E \\ 0, & (v_i, v_j) \notin E \end{cases} . \tag{33}$$

Šetnja W u grafu $G = (V, E)$ je niz

$$W = (v_0, e_1, v_1, e_2, v_2, ..., e_k, v_k) \tag{34}$$

gde su $v_0, v_1, v_2, ..., v_k$ čvorovi grafa G, a e_i je grana čiji su krajevi v_{i-1} i v_i, za $i = 1, 2, ..., k$. **Dužina šetnje** W je k, tj. jednaka je broju grana, s tim što može biti i $k = 0$. Ako je $v_0 = v_k$, onda kažemo da je W **zatvorena šetnja**. U šetnji definisanoj na ovaj način čvorovi i grane se mogu ponavljati. Ukoliko nije dozvoljeno ponavljanje grana, šetnja se zove **staza**. Šetnja u kojoj nema ponavljanja ni čvorova ni grana zove se **put**.

Ciklus u neusmerenom grafu je šetnja $(v_0, e_1, v_1, e_2, v_2, ..., e_k, v_k)$ takva da je $k > 1$, prvih k čvorova se međusobno razlikuju i $v_0 = v_k$. To znači da je ciklus u neusmerenom grafu zatvoren put isključujući prvi i poslednji čvor. Graf bez ciklusa zovemo **aciklički graf**. Graf je **aperiodičan** ako je najveći zajednički delilac dužina svih ciklusa u grafu jednak jedan. Sistematično kretanje od čvora do čvora duž grana grafa tako da se tačno jednom posete svi čvorovi grafa zove se **obilazak grafa**. Ovakav obilazak može se početi iz unapred zadatog čvora ili se početni čvor može izabrati na slučajan način. Redosled obilaska čvorova, osim pravila izbora narednog čvora za obilazak, zavisi i od izbora početnog čvora.

Kažemo da su čvorovi u i v grafa $G = (V, E)$ **povezani** ukoliko postoji šetnja od u do v u grafu G. **Graf je povezan** ako su bilo koja dva čvora povezana.

Ispitivanje povezanosti čvora grafa predstavlja važan algoritamski problem ove teorije. Za čvorove s i t grafa G, pitanje povezanosti se rešava nalaženjem algoritma kojim se utvrđuje postojanje puta između čvorova s i t u grafu G. Može se postaviti i pitanje da li postoji najkraći put između njih. U tu svrhu definišemo rastojanje između čvorova grafa kao dužinu minimalnog puta između njih. Rastojanje između čvorova grafa je beskonačno ukoliko ne postoji put između njih. Rastojanju čvorova grafa ne odgovara fizičko rastojanje između tih čvorova. Graf bez višestrukih grana između istih čvorova i bez petlji zove se **jednostavan graf**. **Potpun graf** je graf u kom postoji grana između svaka dva čvora. **Stablo** je povezani graf sa najmanjim mogućim brojem grana. Za stablo sa $|V|$ čvorova, broj grana u tom stablu je $|E| = |V| - 1$.

Težinski graf (engl. weighted graphs) je uređeni par (G, ω), gde je $G = (V, E)$ graf i

$$\omega : E \to R_0^+ \tag{35}$$

zove se **težinska funkcija**. Vrednost funkcije $\omega(e)$, gde je $e \in E$, zove se **težina grane** e. **Težina grafa** je definisana kao suma težina grana koje obrazuju taj graf. Težinski graf često se naziva i **mreža**. U opštem slučaju kodomen težinske funkcije može biti i neki drugi skup. Neka su čvorovi potpunog težinskog grafa G numerisani sa $1, 2, ..., n$. Grani u između čvorova i i j, dodeljena je težina $\omega(u) = d_{ij}$. Pomoću težina d_{ij} možemo obrazovati kvadratnu matricu $D = [d_{ij}]_{nxn}$. Matrica D se naziva **težinska matrica** ili matrica rastojanja.

Razapinjuće stablo je jednostavan povezan graf bez ciklusa, odnosno razapinjuće stablo je podgraf koji je stablo kod koga su povezani svi čvorovi koji su bili povezani u prvobitnom grafu. Razapinjuće stablo povezanog, težinskog i neusmerenog grafa koje sadrži sve čvorove tog grafa, a zbir težina njegovih grana je minimalan nazivamo **minimalno razapinjuće stablo**. Jedan graf može imati više minimalnih razapinjućih stabala.

Problem trgovačkog putnika je reprezentativan primer problema minimalnog

razapinjućeg stabla. Ukoliko gradovi predstavljaju čvorove, težina grana je fizičko rastojanje gradova.

Ukoliko svakoj grani dodelimo određenu težinu, onda zbir težina grana u razapinjućem stablu predstavlja **težinu tog stabla**. Minimalno razapinjuće stablo nekog grafa je razapinjuće stablo čija je težina manja ili jednaka od težine ostalih razapinjućih stabala tog grafa. Svaki neusmeren graf, koji ne mora biti povezan, ima **minimalnu razapinjuću šumu**, koja je unija minimalnih razapinjućih stabala. Ako svaka grana ima različitu težinu, onda graf sadrži samo jedno jedinstveno minimalno razapinjuće stablo. Ako je neka grana grafa jedinstvena grana minimalne težine, onda se ona nalazi u svim minimalnim razapinjućim stablima tog grafa.

4.2 Osnovni pojmovi

Definišimo prvo pojmove koje ćemo koristiti u ovoj sekciji. Grane težinskog grafa, čiji su čvorovi u istom klasteru, zovemo **unutrašnje grane**, a one čiji su čvorovi u različitim klasterima, zovemo **spoljašnje grane** tih klastera. Može se dogoditi da težina neke spoljašnje grane klastera bude manja od težine neke unutašnje grane tog klastera. Takvu spoljašnju granu zovemo **uljez grana**. Navedimo sada definiciju uljeza nekog čvora.

Neka je dat skup $K = \{x_1, ..., x_n, \hat{x}\}$, čiji su elementi čvorovi težinskog grafa, gde je $K_0 = \{x_1, ..., x_n\}$ klaster. Sa $d(a, b)$ označimo rastojanje čvorova grafa a i b, odnosno težinu grane (a, b). Neka je

$$d = \max\{d(x_i, x_j)\}, x_i, x_j \in K_0. \tag{36}$$

Kažemo da je $\hat{x}$ **uljez u klasteru** K_0 **koji odgovara čvoru** x_k, ako su ispunjeni sledeći uslovi:

$$d(x_k, \hat{x}) < d,\ \hat{x} \neq x_i,\ i = 1, 2, ..., n. \tag{37}$$

Dva faktora utiču na pojavljivanje uljeza:

- Prvo, parovi čvorova u klasteru sa relativno velikim međusobnim rastojanjem,

- Drugo, parovi čvorova, koji nisu u istom klasteru, sa relativno malim međusobnim rastojanjem.

Cilj je da se kreira algoritam za klasterovanje grafova koji je pogodan za grafove sa značajnom međuklasterskom sličnošću. Za meru pomenute sličnosti možemo koristiti broj uljeza. Čvorovi takvog grafa imaju značajan broj uljeza. Kako je broj čvorova u klasterima nepoznat, prikladnije je koristiti procenat nego broj uljeza. U radu razmatramo grafove kod kojih broj uljeza koji odgovaraju čvoru x klastera K_x, može biti najviše $p\%$ od ukupnog broja čvorova klastera K_x. Neki uljezi mogu biti odgovarajući za više čvorova. Takođe, za svaki čvor ostavljamo mogućnost gubitka $p\%$ unutrašnjih grana u toku izvršavanja algoritma. Osnovna ideja je prepoznavanje i uklanjanje svih spoljašnjih grana. Prirodno, prvo ispitujemo najteže grane. Tokom procesa iteracije, uklanjaju se grane, jedna po jedna. Analiziramo celu strukturu, sve zajedničke susedne čvorove čvorova ispitivane grane. Ta struktura, ukoliko je u klasteru, trebalo bi da bude čvrsto povezana, sa ograničenim brojem dodatnih grana (uljez grana) povezanih sa njom.

U sledeće dve podsekcije, razmatramo klasterovanje potpunog težinskog grafa. Prvo predstavljamo novi algoritam za klasterovanje (IBC1), baziran na uljezima, u slučaju kada broj klastera nije unapred dat. Potom, predstavljamo dodatni algoritam (IBC2) u slučaju kada je broj klastera unapred dat.

4.3 Klasterovanje bez pretpostavke o broju klastera

Neka je V skup svih čvorova težinskog grafa G. Ako su čvorovi a i b $(a, b \in G)$ povezani granom, onda tu granu označavamo kao neuređen par (a, b). Neka je

$$k = ((a_1, b_1), (a_2, b_2), ..., (a_m, b_m)) \tag{38}$$

niz grana grafa tako da težine grana obrazuju neopadajući niz (dalje u tekstu, neopadajući niz grana).

Dalje, definišemo preslikavanje

$$\mathcal{F}_k : V \longrightarrow P(V), \tag{39}$$

gde je $P(V)$ partitivni skup skupa V, na sledeći način:

$$\mathcal{F}_k(x) = \{x\} \cup \{x_i \in V \mid (x, x_i) \text{ je grana niza } k \text{ (41)}\}. \tag{40}$$

U nastavku $\mathcal{F}_k(x)$ ćemo označavati sa X. Na taj način svakom čvoru x pridružujemo skup X čiji su elementi taj čvor i svi njegovi susedni čvorovi iz niza grana (41).

Dalje dajemo opis algoritma za klasterovanje potpunog težinskog grafa kada nije poznat broj klastera.

Prvi korak: Prvo formiramo neopadajući niz grana grafa. Zatim izračunamo prosečno rastojanje između svih čvorova grafa (aritmetička sredina težina grana). Smatramo da čvorovi, čije je međusobno rastojanje veće od prosečnog, nisu slični, odnosno ne pripadaju istom klasteru. Čvorovi, čije je rastojanje od svih ostalih čvorova veće od prosečnog, su jednočlani klasteri. Neka je c neopadajući niz grana čija je težina jednaka ili manja od prosečne.

Drugi korak: Prvo biramo najtežu granu, odnosno najboljeg kandidata za spoljašnju granu iz niza grana

$$c = ((a_1, b_1), (a_2, b_2), ..., (a_M, b_M)) \tag{41}$$

.

Razlikujemo tri slučaja:

A) Težina poslednje grane (a_M, b_M) u nizu c je različita od težine ostalih grana niza i ovu granu označimo sa (x, y).

B) Poslednjih r grana u nizu c

$$(a_{M-(r-1)}, b_{M-(r-1)}), ..., (a_M, b_M) \tag{42}$$

imaju istu težinu i

$$card(A_i \cap B_i), \quad (i = M - (r-1), ..., M) \tag{43}$$

ima jedinstvenu najmanju vrednost, gde je $A_i = \mathcal{F}_c(a_i)$ i $B_i = \mathcal{F}_c(b_i)$. Neka ta jedinstvena vrednost odgovara grani koju ćemo označiti sa (x, y).

C) Poslednjih r grana u nizu c, tj. (42) imaju istu težinu i nekoliko (43) ima istu najmanju vrednost. Od grana kojima odgovara ista najmanja vrednost biramo jednu granu koju ćemo označiti sa (x, y).

Neka je (x, y) grana određena u nekom od prethodna tri slučaja. Jasno je da ne mogu istovremeno nastupiti dva slučaja. Neka je

$$\mathcal{F}_c(x) = X = \{x, y, x_1, ..., x_n, \hat{x}_1, \hat{x}_2, ..., \hat{x}_k\} \tag{44}$$

$$\mathcal{F}_c(y) = Y = \{x, y, x_1, ..., x_n, \hat{y}_1, \hat{y}_2, ..., \hat{y}_s\} \tag{45}$$

Neki od skupova $\{x_1, ..., x_n\}$, $\{\hat{x}_1, \hat{x}_2, ..., \hat{x}_k\}$, $\{\hat{y}_1, \hat{y}_2, ..., \hat{y}_s\}$ mogu biti prazni skupovi.

Treći korak: Dalje formiramo skup kandidata za klaster

$$Z = X \cap Y = \{x, y, x_1, ..., x_n\}. \tag{46}$$

Skup Z nije prazan jer sadrži bar čvorove x i y. Čvorovi iz skupova X i Y, koji ne pripadaju Z, ukoliko postoje, su kandidati za uljeze. Dalje razlikujemo dva slučaja:

i) Ako je

$$k > (n+2)\frac{p}{100} \quad \text{ili} \quad s > (n+2)\frac{p}{100}, \tag{47}$$

onda smatramo da čvorovi x i y ne pripadaju istom klasteru. Grana (x, y) se uklanja iz niza. Posle uklanjanja grane (x, y) dobijamo novi niz koji ima jednu granu manje i koji ćemo opet označiti sa c, a zatim sledi povratak na drugi korak.

ii) Ako je

$$k \leqslant (n+2)\frac{p}{100} \quad \text{i} \quad s \leqslant (n+2)\frac{p}{100}. \tag{48}$$

onda proveravamo da li je neka od grana $(x, \hat{y}_j), j = 1, ..., s$, $(y, \hat{x}_i), i = 1, .., k$ unutrašnja, koja je uklonjena u toku izvršavanja algoritma. Za svaki preostali element $\hat{x}_i$ iz X i $\hat{y}_j$ iz Y, ukoliko postoji, proveravamo da li se sadrži u

$(100-p)\%$ skupova pridruženih elementima skupa Z. Svi čvorovi, koji ne ispunjavaju prethodni uslov, su uljezi. Svakom čvoru $\hat{x}_i$, odnosno $\hat{y}_j$, koji ispunjava prethodni uslov pridružujemo skup $\widehat{X}_i = \mathcal{F}_c(\hat{x}_i)$, odnosno $\widehat{Y}_j = \mathcal{F}_c(\hat{y}_j)$. Zatim određujemo skup

$$\begin{aligned} \widehat{Z} &= \left\{\hat{x}_i \mid card(Z \setminus \widehat{X}_i) \leqslant card(Z)\frac{p}{100}\right\} \\ &\bigcup \left\{\hat{y}_j \mid card(Z \setminus \widehat{Y}_j) \leqslant card(Z)\frac{p}{100}\right\} \end{aligned} \tag{49}$$

Novi skup kandidata za klaster je

$$Z^+ = Z \cup \widehat{Z}, \tag{50}$$

gde $\widehat{Z}$ može biti i prazan skup. Zatim za svaki $x_i \in Z^+$ proveravamo koliko je povezan sa ostalim čvorovima iz Z^+. Zato određujemo skupove

$$X^i = X_i \cap Z^+, \tag{51}$$

gde je X_i skup pridružen čvoru $x_i \in Z^+$. Proveravamo preklapanja skupova Z^+ i X^i. Skup Z^+ može sadržati uljeze zajedničke za x i y. Takođe, možda je došlo do ranijeg uklanjanja unutrašnjih grana koje sadrže čvor x_i. Uz to, proveravamo i da li je broj uljeza koji odgovaraju čvoru x_i u dozvoljenim granicama. Provere vršimo na sledeći način. Ako je

$$card(Z^+ \setminus X^i) > card(Z^+)\frac{p}{100} \ \vee \ card(X_i \setminus X^i) > card(X_i)\frac{p}{100}, \tag{52}$$

smatramo da čvor x_i ne pripada klasteru čvora x. Zatim određujemo

$$\begin{aligned} K_X &= \left\{x_i \in Z^+ \mid card(Z^+ \setminus X^i) \leqslant card(Z^+)\frac{p}{100}\right\} \\ &\bigcap \left\{x_i \in Z^+ \mid card(X_i \setminus X^i) \leqslant card(X_i)\frac{p}{100}\right\}. \end{aligned} \tag{53}$$

Osim zajedničkih uljeza za x i y svi ostali elementi iz Z^+, će biti u K_X. Dalje vršimo proveru da li je

$$card(Z^+) - card(K_x) > card(K_x)\frac{p}{100}. \tag{54}$$

Ako važi nejednakost (54), smatramo da K_x nije klaster i da je grana (x, y) spoljašnja. Grana (x, y) se uklanja iz niza. Posle uklanjanja grane (x, y) dobijamo novi niz koji ima jednu granu manje i koji ćemo opet označiti sa c, a zatim sledi povratak na drugi korak.

Ako ne važi nejednakost (54), smatramo da je K_x klaster koji sadrži čvor x. Sve grane koje sadrže neke od čvorova formiranog klastera se uklanjaju iz niza. Posle uklanjanja ovih grana dobijamo novi niz koji ima manje grana i koji ćemo opet označiti sa c. Ukoliko nisu formirani svi klasteri (odnosno nisu uklonjene sve grane iz niza), sledi povratak na drugi korak. Prvi i drugi korak ovog algoritma su bazirani isključivo na distancama između čvorova. Treći korak je baziran na povezanosti čvorova.

4.4 Klasterovanje kada je broj klastera unapred dat

U algoritmu, opisanom u poslednjoj podsekciji, polazi se od pretpostavke da broj klastera nije poznat. Ukoliko je broj klastera poznat, recimo n, prethodnim algoritmom se odrede klasteri koji služe kao polazna osnova (polazni klasteri) za preciznije određivanje finalnih klastera. Pretpostavljamo da klasteri imaju približno isti broj čvorova. Ako je broj polaznih klastera manji od n, opisanim algoritmom se ne mogu odrediti traženi klasteri.

Prvi korak: Ako je broj polaznih klastera jednak n, svi polazni klasteri su ujedno i finalni klasteri. Dalje razmatramo samo slučaj kada je broj polaznih klastera veći od n.

Neka je

$$(K_1, K_2, ..., K_m) \tag{55}$$

niz svih polaznih klastera takav da je

$$(card(K_1), card(K_2), ..., card(K_m)) \tag{56}$$

nerastući niz. Razlikujemo dva slučaja:

1. Postoji $j \in N$ tako da je

$$j = \min_{\substack{n \leqslant i < m \\ card(K_i) \neq card(K_{i+1})}} i \tag{57}$$

U ovom slučaju kandidati za nove polazne klastere su klasteri

$$K_1, K_2, ..., K_j. \tag{58}$$

2. Ako takvo j ne postoji, onda poslednjih $k(k > m-n)$ klastera u nizu (55) imaju isti broj čvorova. Za svaki od poslednjih k klastera iz niza određujemo prosečno rastojanje između svih njegovih čvorova. Kandidati za nove polazne klastere ostaju svi polazni klasteri osim jednog sa najvećim prosečnim rastojanjem.

Drugi korak: Dalje, za svaki čvor polaznih klastera, koji nisu kandidati posle prvog koraka, određujemo prosečno rastojanje od svih čvorova svakog kandidata za novi polazni klaster posebno. Ako posmatrani čvor ima najmanje prosečno rastojanje od čvorova jednog ili više kandidata, na kraju provere taj čvor dodajemo jednom od tih kandidata. Posle ovog koraka svi kandidati postaju novi polazni klasteri. Sledi povratak na prvi korak.

5 Teorijska osnova za bradu signala

Da bi se opisani algoritmi primenili na klasterovanje ljudskih aktivnosti, prvo je potrebno modeliranjem prostorno-vremenskih signala dobiti simbole i pomoću tih simbola formirati sekvence stringova. Sekvence stringova mogu se shvatiti kao čvorovi težinskog grafa, pri čemu se za težine grana mogu uzeti modifikovana Levenštajnova rastojanja.

5.1 Teorijska osnova za formiranje sekvenci stringova

Kao teorijska osnova za formiranje sekvenci stringova može poslužiti sledeće razmatranje.

Neka su dati skupovi $I_k = \{1, 2, ..., k\}, J_n = \{1, 2, ..., n\}$, gde je $k, n \in N$ i $k, n \geqslant 2$. Na skupu $I_k \times J_n$ definišemo preslikavanje

$$p : I_k \times J_n \longrightarrow R, \tag{59}$$

pri čemu je $p(i, j) = p_{ij} \in R$. Pomoću preslikavanja p određena je matrica

$$P = \begin{pmatrix} p_{11} & p_{12} & \dots & p_{1n} \\ p_{21} & p_{22} & \dots & p_{2n} \\ \dots & \dots & \dots & \dots \\ p_{k1} & p_{k2} & \dots & p_{kn} \end{pmatrix} \tag{60}$$

Na skupu $I_k \times J_n$ definišemo relcije M i m na sledeći način:

$$i\ M\ j \Longleftrightarrow \begin{cases} p_{ij} > p_{(i+1)j} & \text{ako} \quad i = 1 \\ p_{ij} > p_{(i-1)j} & \text{ako} \quad i = n \\ p_{(i-1)j} < p_{ij} \ \wedge \ p_{ij} > p_{(i+1)j} & \text{ako} \quad 1 < i < n, \end{cases} \tag{61}$$

$$i\ m\ j \Longleftrightarrow \begin{cases} p_{ij} < p_{(i+1)j} & \text{ako} \quad i = 1 \\ p_{ij} < p_{(i-1)j} & \text{ako} \quad i = n \\ p_{(i-1)j} > p_{ij} \ \wedge \ p_{ij} < p_{(i+1)j} & \text{ako} \quad 1 < i < n. \end{cases} \tag{62}$$

Neposredno je jasno da je tačno sledeće tvrđenje

$$(\forall i, j)\neg(i\ M\ j \ \wedge \ i\ m\ j). \tag{63}$$

Koristeći uvedene relacije definišemo preslikavanje

$$f : I_k \times J_n \longrightarrow S, \tag{64}$$

gde je

$$S = \{M_1, m_1, M_2, m_2, ..., M_n, m_n, \varepsilon\} \tag{65}$$

skup simbola, na sledeći način

$$f(i,j) = \begin{cases} M_j & \text{ako} \quad \tau(i\ M\ j) = \top \\ m_j & \text{ako} \quad \tau(i\ m\ j) = \top \\ \varepsilon & \text{otherwise.} \end{cases} \tag{66}$$

Pomoću preslikavanja f je određena matrica

$$F = \begin{pmatrix} f(1,1) & f(1,2) & \dots & f(1,n) \\ f(2,1) & f(2,2) & \dots & f(2,n) \\ \dots & \dots & \dots & \dots \\ f(k,1) & f(k,2) & \dots & f(k,n) \end{pmatrix}. \tag{67}$$

Definišimo još preslikavanje

$$\varphi : F \longrightarrow (S_1, S_2, ..., S_k)^T \tag{68}$$

tako da je

$$S_i = f(i,1) \oplus f(i,2) \oplus \dots \oplus f(i,n), \tag{69}$$

gde je $\oplus$ konkatenacija. Tako dobijamo sekvencu stringova čiji su simboli iz skupa S.

Na kraju u svakom stringu izostavljamo simbol ε i sve stringove u kojima, osim ε, nije bilo drugih simbola. Tako dobijamo sekvencu stringova

$$a = (\widehat{S}_1, \widehat{S}_2, ..., \widehat{S}_{\hat{k}})^T \tag{70}$$

pri čemu je $\hat{k} \leqslant k$.

5.2 Modifikovano Levenštajnovo rastojanje

Preslikavanje

$$a : \{1, 2, ..., n\} \to S \tag{71}$$

zovemo **konačan niz** (u daljem tekstu **niz**). Ako je $S = R$, gde je R skup realnih brojeva, onda takav niz zovemo **brojni niz**. Ako je S skup simbola neke azbuke, onda ćemo takav niz zvati **string** ili tekstualni string. Dalje ćemo, kada kažemo string, podrazumevati skup slika $a(1)a(2)\cdots a(n)$ ili kraće $a_1a_2\cdots a_n$.

Za definisanje mere sličnosti između stringova koristi se pojam rastojanja između stringova. U tu svrhu su uvedene operacije sa znakovima stringova: **umetanje znaka, brisanje znaka i zamena jednog znaka drugim**. Minimalan broj prethodno uvedenih operacija, potrebnih da se od jednog stringa dobije drugi string, zove se **Levenštajnovo rastojanje**, (Levenshtein, 1966). Levenštajnovo rastojanje predstavlja meru sličnosti između nizova znakova koji mogu biti iste ili različite dužine. Ako je rastojanje između stringova a i b manje od rastojanja stringova c i d, onda kažemo da su stringovi a i b sličniji nego stringovi c i d. Takođe, možemo reći da svaka od uvedenih operacija: dodavanje znaka na proizvoljno mesto, brisanje znaka i zamenu jednog znaka drugim, ima težinu (cenu) jedan. Levenštajnovo rastojanje predstavlja minimalnu ukupnu cenu transformacije jednog stringa u drugi primenom navedenih operacija proizvoljnim redosledom. Dakle, polazi se od pretpostavke da navedene operacije jednako doprinose rastojanju između stringova, pa je samo potrebno odrediti minimalan broj operacija potrebnih da se jedan string transformiše u drugi.

U opštem slučaju, ove operacije mogu imati različite težine (cene) u zavisnosti od oblasti primene. Ponekad se težina operacije zamene definiše kao zbir težina operacija brisanja i umetanja.

U nastavku predstavljamo algoritam za računanje Levenštajnovog rastojanja.

Neka su dati stringovi S_a i S_b čije su dužine $|S_a|$ i $|S_b|$. String S_a ćemo zvati izvorni string, a S_b odredišni string. Primenom algoritma se dobija minimalni broj operacija potrebnih da se izvorni string transformiše u odredišni,

odnosno vrednost Levenštajnovog rastojanja ovih stringova. Za stringove S_a i S_b se formira matrica rastojanja D dimenzija $(|S_a|+1) \times (|S_b|+1)$. Svakom znaku izvornog stringa S_a odgovara jedna vrsta, a svakom znaku odredišnog stringa S_b jedna kolona matrice D. Prva vrsta i prva kolona matrice D se formiraju na sledeći način:

$$\begin{aligned} &(\forall i)(0 \leqslant i \leqslant |S_a|)\ D(i,0) = i, \\ &(\forall j)(0 \leqslant j \leqslant |S_b|)\ D(0,j) = j. \end{aligned} \tag{72}$$

Ostale vrste i kolone se rekurzivno formiraju koristeći obrazac:

$$\begin{aligned} &(\forall i)(\forall j)(1 \leqslant i \leqslant |S_a|)(1 \leqslant j \leqslant |S_b|) \\ &D(i,j) = min \begin{cases} D(i-1,j)+1, \\ D(i,j-1)+1, \\ D(i-1,j-1) + \begin{cases} 1, \ if \ \ S_a(i-1) \neq S_b(j-1) \\ 0, \ if \ \ S_a(i-1) = S_b(j-1) \end{cases} \end{cases} \end{aligned} \tag{73}$$

Traženo Levenštajnovo rastojanje se određuje na sledeći način:

$$\lambda(S_a, S_b) = D(|S_a|, |S_b|), \tag{74}$$

Ako težine operacija imaju težinu različitu od 1, dobijamo uopšteno Levenštajnovo rastojanje, čiji algoritam sledi u nastavku rada.

Prva vrsta i prva kolona matrice D se formiraju na sledeći način:

$$\begin{aligned} &D(0,0) = 0, \\ &(\forall i)(0 < i \leqslant |S_a|)\ D(i,0) = D(i-1,0) + \text{težina brisanja znaka } S_a[i], \\ &(\forall j)(0 < j \leqslant |S_b|)\ D(0,j) = D(0,j-1) + \text{težina umetanja znaka } S_b[j]. \end{aligned} \tag{75}$$

Ostale vrste i kolone se rekurzivno formiraju koristeći obrazac:

$$(\forall i)(\forall j)(1 \leqslant i \leqslant |S_a|)(1 \leqslant j \leqslant |S_b|)$$

$$D(i,j) = min \begin{cases} D(i-1,j) + \text{težina brisanja znaka } S_a[i], \\ D(i,j-1) + \text{težina umetanja znaka } S_b[j], \\ D(i-1,j-1) + \text{težina zamene znaka } S_a[i] \text{ znakom } S_b[j]. \end{cases} \tag{76}$$

Navodimo iterativni algoritam za računanje Levenštajnovog rastojanja:

---

function minimalno_rastojanje (S_a, S_b)

m = dužina (S_a)

n = dužina (S_b)

D = matrica dimenzija $(m+1) \times (n+1)$

$D(0,0) = 0$

for $i = 1$ **to** m **do**

  D[i,0] = D[i-1, 0] + težina_brisanja ($S_a[i]$)

for $j = 1$ **to** n **do**

  D[0,j] = D[0, j-1] + težina_umetanja ($S_b[j]$)

for $i = 1$ **to** m **do**

  for $j = 1$ **to** n **do**

    D(i,j) = min (D[i-1,j] + težina_brisanja ($S_a[i]$),

                  D[i,j-1] + težina_umetanja ($S_b[j]$),

                  D[i-1,j-1] + težina_zamene ($S_a[i], S_b[j]$))

return D[m,n];

Ako su relativno velike razlike u dužini stringova, pogodnije je umesto Levenštajnovog koristiti normalizovano Levenštajnovo rastojanje:

$$\lambda_N(S_a, S_b) = \frac{\lambda(S_a, S_b)}{|S_a| + |S_b|}. \tag{77}$$

Na kraju ove podsekcije navodimo adaptirano Levenštajnovo rastojanje kojim se određuje rastojanje između dve sekvence stringova. Težine operacija su date

na sledeći način:

1. Težine umetanja ili brisanja stringa jednaki su njegovoj dužini.
2. Težine zamene jednog stringa drugim jednaki su standardnom Levenštajnovom rastojanju između njih.

Adaptirano Levenštajnovo rastojanje između dve sekvence stringova, a_1 i a_2 definišemo na sledeći način:

$$\hat{\lambda}(a_1, a_2) = \frac{\widehat{D}(|a_1|, |a_2|)}{|a_1| + |a_2|}, \tag{78}$$

gde je:

$$\begin{aligned} &\widehat{D}(0,0) = 0, \\ &(\forall i)(1 \leqslant i \leqslant |a_1|)\ \widehat{D}(i,0) = \widehat{D}(i-1,0) + |a_1(i-1)|, \\ &(\forall j)(1 \leqslant j \leqslant |a_2|)\ \widehat{D}(0,j) = \widehat{D}(0,j-1) + |a_2(j-1)|, \end{aligned} \tag{79}$$

i

$$(\forall i)(\forall j)(1 \leqslant i \leqslant |a_1|)(1 \leqslant j \leqslant |a_2|) \tag{80}$$

$$\widehat{D}(i,j) = min \begin{cases} \widehat{D}(i-1,j) + |a_1(i-1)|, \\ \widehat{D}(i,j-1) + |a_2(j-1)|, \\ \widehat{D}(i-1,j-1) + \lambda(a_1(i-1), a_2(j-1)). \end{cases} \tag{81}$$

5.3 Formiranje sekvenci stringova

Različiti parametri (npr. ugaona brzina, trenutno ubrzanje, apsolutna orijentacija), čije se vrednosti dobijaju pomoću mernih jedinica, koriste se za analizu ljudskih aktivnosti. Podaci iz mernih jedinica, poređani po vremenskom redosledu, mogu biti prikazani u obliku matrice P (60). Element $p_{ij},\ i = 1, 2, ..., k;\ j = 1, 2, ..., n,$ matrice P je vrednost parametra sa rednim brojem j u vremenskoj tački i. Ovih n parametara imaju svoje lokalne ekstreme (maksimuma i minimuma). Opisana je procedura za dodeljivanje simbola ovim ekstremima. U posmatranom trenutku, nekoliko parametara može imati ekstremne vrednosti. Od svih simbola dodeljenih ekstremima u posmatranoj vremenskoj tački, formira se string. Stringovi poređani u vremenskom redosledu čine sekvencu stringova.

Opisanim modeliranjem signala iz merenja na osnovu simbola jedinice, umesto znakova dobijaju se stringovi, a umesto stringa sekvenca stringova. Zbog toga je bila potrebna adaptacija operacija koje se koriste prilikom računanja Levenštajnovog rastojanja. Adaptirano Levenštajnovo rastojanje (78) korišćeno u ovom pristupu kvantifikuje sličnost između dve sekvence stringova.

6 Evaluacija rezultata klasterovanja

Kao rezultat primene nekog algoritma klasterovanja se dobijaju klasteri koji nisu unapred poznati, pa je, iz tog razloga, poželjan neki oblik procene rezultata klasterovanja. Dakle, po završenom klasterovanju uvek se postavlja pitanje da li je to klasterovanje dobro ili loše. Od odgovora na ovo pitanje zavisi da li uopšte ima smisla koristiti predloženi algoritam. Odgovor na ovo pitanje nije jednostavan i nije lako dobiti pouzdan odgovor. Naime, ne postoji precizan odgovor na pitanje šta je dobro klasterovanje. Često je procena subjektivan sud istraživača. To nije razlog da ne koristimo neke od postojećih mera evaluacije. Svaki istraživač se opredeljuje za one koje smatra najprikladnijim.

6.1 Osnovne mere za evaluaciju rezultata klasterovanja

Metode za evaluaciju rezultata klasterovanja se još zovu i metode klaster validacije. Mogli bismo da kažemo da se validacijom određuje stepen slaganja podele datog skupa i date strukture podataka.

Istraživači su razvili nekoliko kriterijuma za evaluaciju klasterovanja. Ove kriterijume možemo podeliti u dve klase:

- Unutrašnji ili interni kriterijumi,
- Spoljašnji ili eksterni kriterijumi.

6.1.1 Unutrašnji ili interni kriterijumi

Prilikom primene internih mera za evaluaciju rezultata klasterovanja nisu potrebne dodatne informacije o podacima koji se koriste. Koriste se samo informacije dobijene postupkom klasterovanja kao npr.: homogenost klastera, razdvojenost klastera, slaganje ulaznih podataka i rezultata klasterovanja. Najčešće korišćene interne mere su sledeće:

- **Calinski-Harabaszov indeks (CH)** je unutrašnja mera kvaliteta klasterovanja i računa se pomoću obrasca

$$CH = \frac{SS_B}{SS_W} \times \frac{n-k}{k-1}, \tag{82}$$

gdje je n broj objekata, a k broj klastera. SS_B je ukupna varijansa između klastera koja se određuje pomoću obrasca

$$SS_B = \sum_{i=1}^{k} |C_i| \|c_i - m\|^2, \tag{83}$$

gde je C_i kardinalni broj i-tog klastera, m aritmetička sredina svih objekata u skupu, a $\|c_i - m\|$ Euklidsko rastojanje centra i-tog klastera od m.

SS_W je ukupna varijansa objekata unutar klastera i određuje pomoću obrasca

$$SS_W = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2, \tag{84}$$

gde je C_i i-ti klaster, a c_i njegov centar. Razdvojenost klastera je direktno proporcionalna sa veličinom indeksa CH. Ovaj indeks se relativno brzo određuje.

- **Davies-Bouldinov indeks (DB)** je unutrašnja mera kvaliteta klasterovanja koja daje sličnost između svakog klastera C_i, $i = 1, 2, ..., k$ i njemu najsličnijeg klastera C_j. Računa se pomoću obrasca

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} R_{ij}, \tag{85}$$

gde je k broj klastera. R_{ij} je mera sličnosti i-tog i j-tog klastera. Vrednost mere sličnosti R_{ij} se određuje pomoću obrasca

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}, \tag{86}$$

gde je s_i srednje rastojanje svakog objekta i-tog klastera od njegovog centra, a d_{ij} rastojanje centara i-tog i j-tog klastera. Razdvojenost klastera je obrnuto proporcionalna sa veličinom indeksa DB.

- **Koeficijent siluete (S)** je unutrašnja mera kvaliteta klasterovanja i određuje se u tri koraka. U prvom koraku računa se koeficijent siluete svih objekata posebno. U drugom koraku određuje se koeficijent siluete klastera. U trećem koraku određuje se koeficijent siluete S..

  Definišimo prvo pojmove koji se pojavljuju u obrascu za izračunavanje koeficijenta siluete.

  Neka je

  $$X = \{x_1, x_2, ..., x_m\} \tag{87}$$

  dati skup objekata i neka je

  $$\{A^1, A^2, ..., A^k\} \tag{88}$$

  skup svih klastera skupa X. Sa

  $$A^j = \{x_1^j, x_2^j, ..., x_{m_j}^j\} \tag{89}$$

  označimo j-ti klaster, pri čemu je $j = 1, 2, ..., k$ i m_j kardinalni broj klastera A^j. Srednje rastojanje između i-tog objekta klastera A^j i svih ostalih objekata tog klastera, u oznaci a_i^j, određuje se po obrascu

  $$a_i^j = \frac{1}{m_j - 1} \sum_{s=1}^{m_j} d(x_i^j, x_s^j) \ i = 1, 2, ..., m_j, \tag{90}$$

  gde je $d(x_i^j, x_s^j)$ rastojanje između i-tog i s-tog objekta klastera A^j.

  Najmanje srednje rastojanje i-tog objekta klastera A^j od svih objekata klastera A^s, pri čemu je $s = 1, 2, ..., k$ i $s \neq j$, računa se koristeći obrazac

  $$b_i^j = \min_{\substack{l=1,...,k \\ i \neq j}} \left\{ \frac{1}{m_l} \sum_{s=1}^{m_l} d(x_i^j, x_s^l) \right\}, \tag{91}$$

  gde je $i = 1, 2, ..., m_j$.

  Koeficijent siluete i-tog objekta klastera A^j određuje se pomoću obrasca

  $$s_i^j = \frac{b_i^j - a_i^j}{\max\{a_i^j, b_i^j\}}. \tag{92}$$

  Dalje računamo koeficijent siluete klastera A^j

  $$S_j = \frac{1}{m_j} \sum_{i=1}^{m_j} s_i^j. \tag{93}$$

Pomoću koeficijenata siluete klastera računamo koeficijent siluete skupa S:

$$S = \frac{1}{k} \sum_{j=1}^{k} S_j. \tag{94}$$

Koeficijent siluete je broj iz intervala $[-1, 1]$. Pomoću ovog koeficijenta dobijamo informacije o gustini i razdvojenosti klastera. Ukoliko je vrednost koeficijenta veća, razdvojenost klastera je bolja. Ovaj koeficijent nije pogodan za skupove sa velikim brojem objekata.

6.1.2 Spoljašnji ili eksterni kriterijumi

Prilikom primene eksternih mera za evaluaciju rezultata klasterovanja procenjuje se stepen slaganja unapred datih klastera sa klasterima dobijenih postupkom klasterovanja. Najveći nedostatak ovih mera je što u praksi najčešće unapred poznate informacije ne postoje.

Navedimo prvo pojmove koje ćemo koristiti u ovoj podsekciji. Neka je dat skup objekata

$$X = \{x_1, x_2, ..., x_n\} \tag{95}$$

i neka je

$$C = \{C_1, C_2, ..., C_m\} \tag{96}$$

skup svih datih klastera skupa X. Takođe, neka je

$$K = \{K_1, K_2, ..., K_s\} \tag{97}$$

skup svih izvedenih klastera, tj. skup klastera dobijenih postupkom klasterovanja.

Sa TP označimo broj parova objekata koji se nalaze u istom klasteru iz skupa C i u istom klasteru iz skupa K. Te parove objekata ćemo zvati **tačno pozitivni**.

Sa TN označimo broj parova objekata koji se nalaze u različitim klasterima iz skupa C i u različitim klasterima iz skupa K. Te parove objekata ćemo zvati **tačno negativni**.

Sa LN označimo broj parova objekata koji se nalaze u istom klasteru iz skupa C i u različitim klasterima iz skupa K. Te parove objekata ćemo zvati **lažno negativni**.

Sa LP označimo broj parova objekata koji se nalaze u različitim klasterima iz skupa C i u istom klasteru iz skupa K. Te parove objekata ćemo zvati **lažno pozitivni**.

Najčešće korišćene eksterne mere su sledeće:

- **Rand indeks (RI)** je spoljašnja mera kvaliteta klasterovanja, a računa se po obrascu

$$RI = \frac{TP + TN}{TP + FP + FN + TN} = \frac{TP + TN}{\binom{n}{2}} \tag{98}$$

  Rand indeks uzima vrednosti iz intervala $[0, 1]$ i veća vrednost ukazuje na bolji kvalitet klasterovanja.

- **Preciznost (P)** je spoljašnja mera kvaliteta klasterovanja, a računa se po obrascu

$$P = \frac{TP}{TP + FP} \tag{99}$$

  Preciznost uzima vrednosti iz intervala $[0, 1]$ i veća vrednost ukazuje na bolji kvalitet klasterovanja.

- **Odziv (R)** je spoljašnja mera kvaliteta klasterovanja, a računa se po obrascu

$$R = \frac{TP}{TP + FN} \tag{100}$$

  Odziv uzima vrednosti iz intervala $[0, 1]$ i veća vrednost ukazuje na bolji kvalitet klasterovanja.

- **Balansirana F-mera (F)** je spoljašnja mera kvaliteta klasterovanja, a računa se po obrascu

$$R = \frac{2PR}{P + R} \tag{101}$$

  Balansirana F-mera uzima vrednosti iz intervala $[0, 1]$ i veća vrednost ukazuje na bolji kvalitet klasterovanja.

- **Čistoća (Purity)** je veoma jednostavna spoljašnja mera kvaliteta klasterovanja, a određuje se na sledeći način:

Za svaki izvedeni klaster $K_i,\ i = 1, 2, ..., s$ prvo se odredi iz kog od datih klastera ima najviše objekata u klasteru K_i, a zatim broj tih objekata. Zbir tako dobijenih brojeva se podeli sa ukupnim brojem objekata u svim datim klasterima. Dobijeni količnik se zove čistoća.

Nedostatak ove mere je što, u slučaju velikog broja izvedenih klastera, može dati visoku čistoću. Na primer, mera čistoće je 1 ako je svaki izvedeni klaster jednočlan.

6.2 Nova mera evaluacije rezultata klasterovanja

U ovoj podsekciji uvodimo novu meru spoljašnje evaluacije, **Tačnost formiranih klastera**.

Neka su

$$C_1, C_2, ..., C_n \tag{102}$$

svi klasteri potpunog težinskog grafa i neka je

$$\{K_1, K_2, ..., K_n\} \tag{103}$$

kolekcija skupova definisanih na sledeći način:

1. Ako je postupkom klasterovanja dobijen izvedeni klaster koji sadrži više od 50% čvorova klastera C_i i ne sadrži više od 50% čvorova nekog drugog klastera C_j, onda je K_i jednako tom izvedenom klasteru.

2. Ako je postupkom klasterovanja dobijen izvedeni klaster koji sadrži više od 50% čvorova klastera C_i, više od 50% čvorova nekog drugog klastera C_j i $i \leqslant j$, onda je K_i jednako izvedenom klasteru i $K_j = \emptyset$.

   Dalje razmatramo samo izvedene klastere koji ne ispunjavaju uslove navedene u slučajevima 1. i 2.

3. Ukoliko postoji, prvo se odredi izvedeni klaster koji sadrži 50% čvorova klastera C_i sa najmanjim indeksom i. Tada je K_i jednako izvedenom klasteru. Postupak se nastavlja sa preostalim datim i izvedenim klasterima.

4. Skupovi K_i, koji nisu dobijeni prethodnim postupkom, su prazni skupovi.

Ako je K_i neprazan skup, onda kažemo da je formiran klaster koji odgovara C_i. Tačnost formiranih klastera T određuje se na sledeći način:

$$T = \frac{\sum_{i=1}^{n} card(C_i \cap K_i)}{\sum_{i=1}^{n} card(C_i)} \cdot 100\% \tag{104}$$

Predstavljena evaluaciona mera T (24) daje procenat pripadnosti zajedničkih čvorova datih i odgovarajućih formiranih klastera od ukupnog broja svih čvorova klastera koji su unapred dati. Spoljašnja evaluaciona mera T ima određene sličnosti sa Čistoćom, ali je preciznija jer eliminiše mogućnost lažne čistoće.

Mere (25) - (28) se zasnivaju na parovima čvorova i stoga nisu direktno uporedive sa predloženom merom T. Ako parovi čvorova iz istog datog klastera pripadaju različitim izvedenim klasterima, tada se smatra da su ti parovi tačno pozitivni. Po analogiji sa predloženim pristupom, tačno pozitivni parovi čvorova su samo oni tačno pozitivni parovi koji pripadaju formiranim klasterima.

7 Rezultati eksperimenta i komparativna analiza

Da bi se mogao primeniti predloženi algoritam, potrebni su simboli dobijeni modeliranjem prostorno-vrmenskih signala, koji su potom korišćeni za formiranje sekvenci stringova. Sekvence stringova mogu biti shvaćene kao čvorovi težinskog grafa, gde težina grane predstavlja modifikovano Levenštajnovo rastojanje između njenih čvorova.

7.1 Baze podataka i preprocesiranje

U svrhu evaluacije predloženog pristupa u realnom okruženju, korišćene su Carnegie Mellon University Multimodal Activity (CMU-MMAC) [56] i RealWorld [91] baze podataka. Prva baza podataka predstavlja kompleksne a druga jednostavne ljudske aktivnosti.

CMU-MMAC baza podataka sadrži zapise ljudskih subjekata dok pripremaju razna jela u kuhinji. Jedan od modaliteta u ovoj bazi podataka zabeležen je pomoću troosnog inercijalnog mernog uređaja (MicroStrains 3DM-GX1), koji sadrži akcelerometar, žiroskop i magnetometar, koji omogućavaju mere trenutnog ubrzanja, ugaone brzine i apsolutne orijentacije. Signali su žiro-stabilizovani i zabeleženi sa frekvencijom od 125 Hz. Razmatrano je dvanaest ljudskih subjekata (S40, S41, S42, S43, S44, S45, S50, S51, S52, S53, S54, S55), od kojih je svaki snimljen dok priprema četiri različita jela, brauni kolač (brownie), kajgana (eggs), pica (pizza) i sendvič (sandwich), odnosno vrše četiri različite aktivnosti. Jedan od ovih 48 subjekat-aktivnost parova, tj. subjekat S52 koji pravi brauni kolač (S52, brownie) sadrži greške u podacima, stoga je isključen iz razmatranja. Svi razmatrani subjekti su desnoruki, stoga je odlučeno da se posmatraju podaci prikupljeni sa inercijalnog uređaja postavljenog na zglob desne ruke. Na nivou signala, posebno su razmatrane ugaona brzina i trenutno ubrzanje duž tri ose (ukupno šet parametara). Kao ilustracija, vrednosti parametara uzorkovanih u pet uzastopnih vremenskih trenutaka, koji predstavljaju mali fragment subjekat-

aktivnost para (S51,eggs), date su u Tabeli 1. Lista subjekat-aktivnost parovaje data u prvoj koloni Tabele 2.

Selektovani podaci su preprocesirani u dva aspekta. Pre svega, segmenti podataka koji nisu relevantni za izvršenu aktivnost subjekata su isključeni iz razmatranja. U Tabeli 2 je prikazano početno i završno sistemsko vreme relevantnih segmenata aktivnosti, za svaki subjekat-aktivnost par korišćen za testiranje. U cilju efikasnosti, podaci su uzorkovani sa frekvencijom od 1.25 Hz.

Tabela 1: Ilustarcija ulaznih podataka sa inercijalnog mernog uređaja

Acceleration			**Angular Velocity**				
a_x	a_y	a_z	**Roll**	**Pitch**	**Yaw**	**Count**	**System Time**
0.003632	−0.500534	−0.183935	−0.005648	−0.555695	−0.629118	27683	10:04:02:440
0.041871	−0.545610	−0.146977	0.278318	−0.599623	−0.668340	27684	10:04:02:448
0.064516	−0.616962	−0.1523187	0.566050	−0.675871	−0.678695	27685	10:04:02:456
0.085452	−0.758171	−0.182867	0.872607	−0.746784	−0.638845	27686	10:04:02:464
0.096561	−0.878231	−0.172826	1.147788	−0.850643	−0.479762	27687	10:04:02:472

Tabela 2: Subjekat-aktivnost parovi i relevantni segmenti aktivnosti

Subjekat, aktivnost	Početak brojača	Početak sistemskog vremena	Kraj brojača	Kraj sistemskog vremena
S50, brauni	2824	16:39:56:000	51077	16:46:22:000
S50, kajgana	2953	15:42:40:000	36453	15:47:08:000
S50, pica	3312	15:24:22:000	61074	15:32:04:000
S50, sendvič	3059	16:27:46:000	24934	16:30:41:000
S51, brauni	3113	10:39:47:000	45112	10:45:23:000
S51, kajgana	6874	10:01:16:000	34500	10:04:57:000
S51, pica	4151	09:32:28:005	64901	09:40:34:005
S51, sendvič	5056	10:25:04:003	21804	10:27:18:003
S52, kajgana	2627	15:01:01:000	28752	15:04:30:000
S52, pica	5168	14:49:46:000	41543	14:54:37:000
S52, sendvič	2590	15:18:54:006	17090	15:20:50:006
S53, brauni	2487	10:31:06:003	39738	10:36:04:003
S53, kajgana	2318	10:05:07:007	28568	10:08:37:007
S53, pica	1785	09:42:13:005	58917	09:49:50:005
S53, sendvič	2472	10:25:11:000	18347	10:27:18:000
S54, brauni	3785	11:56:08:000	52785	12:02:40:000
S54, kajgana	1715	11:29:03:003	31340	11:33:00:003
S54, pica	3064	11:15:27:000	62064	11:23:19:000
S54, sendvič	2921	11:47:33:000	24672	11:50:27:000
S55, brauni	5205	13:20:55:000	45081	13:26:14:000
S55, kajgana	2934	12:47:49:000	32309	12:51:44:000
S55, pica	3544	12:33:04:000	72297	12:42:14:000
S55, sendvič	3407	13:11:43:006	23657	13:14:25:006

RealWorld baza podataka sadrži zabeleške 15 ljudskih subjekata (starsti 31.9 ± 12.4, visine 173.1 ± 6.9, težine 74.1±13.8, osam osoba muškog i sedam osoba ženskog pola) koji obavljaju fundamentalne aktivnosti, od kojih su razmatrane četiri: hodanje (walking), penjanje uz stepenice (climbing up the stairs), skakanje (jumping)i trčanje (running). Korišćena su dva modaliteta iz ove baze podataka - troosni akcelerometar i žiroskop. Signali su snimljeni sa frekvencijom od 50 Hz. Zbog prirode analiziranih aktivnosti, odlučeno je da se posmatraju signali prikupljeni sa nosivog uređaja postavljenog na potkolenicu subjekata. Na nivou signala, razmatrane su ugaona brzina i trenutno ubrazanje duž tri ose (ukupno šest parametara). Zbog efikasnosti, uzorci su izabrani tako da sve aktivnosti traju 90 sekundi. Dodatno, podaci su uzorkovani sa frekvencijom do 2.5 Hz.

7.2 Rezultati

Subjekat-aktivnost parovi mogu se interpretirati kao čvorovi težinskog grafa. Svakom subjekat-aktivnost paru odgovara jedna sekvenca stringova. Skup čvorova koji odgovara istoj aktivnosti formira klaster. Da bi se izvršilo klasterovanje potreban je ulazni parametar p, koji predstavlja maksimalan procenat uljeza, koji je nepoznat.

Prvo su prezentovani rezultati dobijeni korišćenjem CMU-MMAC baze podataka. Recepti za pripremanje jela nisu precizno definisani, stoga različite individue pripremaju ista jela na različite načine. Takođe, postoje segmenti radnji koji su veoma slični pri pripremanju različitih jela. Zasnovano na prethodnom, razumno je pretpostaviti da procenat uljeza nije mali. Da bismo preciznije odredili parametar p, odnosno gornju granicu procenta uljeza, skup subjekata podeljen je na dva disujunktna podskupa. Prvi podskup je S40, S41, S42, S43, S44, S45, a drugi je S50, S51, S52, S53, S54, S55. Prvi podskup, koji sadrži 24 subjekat-aktivnost para, korišćen je za treniranje. Drugi podskup, koji sadrži 23 subjekat-aktivnost para, korišćen je za testiranje. Tokom treniranja, kriterijum tačnost formiranih klastera je korišćen je za određivanje parametra p. Po ovom kriterijumu dobijena je vrednost $p = 30$, koja je u očekivanom opsegu. Koristeći

iste uzorke, izvršena su dva testa. Prvi test je sproveden koristeći IBC1 algoritam (podsekcija 4.3) bez pretpostavke o broju klastera. Potom je izvršeno testiranje na istom uzorku koristeći IBC2 algoritam (podsekcija 4.4), kada je broj klastera unapred dat, tj. za posmatrani uzorak je 4. Rezultati klasterovanja bez pretpostavke o broju klastera su dati u Tabeli 3, dok su rezultati klasterovanja u slučaju kada je broj klastera unapred dat prikazani u Tabeli 4. Da bi evaluacija bila potpuna, izračunati su Rand Indeks (Rand index), preciznost (precision), odziv (recall) i balansirana F-mera (balanced F-measure). Vrednosti parametara evaluacije prikazani su u Tabeli 5 za slučaj bez pretpostavke o broju klastera, i Tabeli 6 kada je broj klastera unapred dat.

Tabela 3: Rezultati klasterovanja dobijeni pomoću IBC1 algoritma koristeći CMU-MMAC bazu podataka

Subjekat, aktivnost	Klasteri					
	K_1	K_2	K_3	K_4	K_5	K_6
S50, brauni	•					
S51, brauni	•					
S53, brauni					•	
S54, brauni	•					
S55, brauni	•					
S50, kajgana		•				
S51, kajgana		•				
S52, kajgana		•				
S53, kajgana		•				
S54, kajgana		•				
S55, kajgana		•				
S50, pica			•			
S51, pica			•			
S52, pica					•	
S53, pica			•			
S54, pica			•			
S55, pica			•			
S50, sendvič						•
S51, sendvič				•		
S52, sendvič				•		
S53, sendvič				•		
S54, sendvič				•		
S55, sendvič				•		

Tabela 4: Rezultati klasterovanja dobijeni pomoću IBC2 algoritma koristeći CMU-MMAC bazu podataka

Subjekat, aktivnost	Klasteri			
	K_1	K_2	K_3	K_4
S50, brauni	•			
S51, brauni	•			
S53, brauni	•			
S54, brauni	•			
S55, brauni	•			
S50, kajgana		•		
S51, kajgana				
S52, kajgana		•		
S53, kajgana		•		
S54, kajgana		•		
S55, kajgana		•		
S50, pica			•	
S51, pica			•	
S52, pica	•			
S53, pica			•	
S54, pica			•	
S55, pica			•	
S50, sendvič				•
S51, sendvič				•
S52, sendvič				•
S53, sendvič				•
S54, sendvič				•
S55, sendvič				•

Tabela 5: Evaluacija rezultata klasterovanja dobijenih pomoću IBC1 algoritma koristeći CMU-MMAC bazu podataka

Tačnost formiranih klastera		$T = 86.96\%$	
Tačno pozitivni	TP=41	Rand indeks	RI=0.941
Tačno negativni	TN=197	Preciznost	P=0.976
Lažno pozitivni	FP=1	Odziv	R=0.745
Lažno negativni	FN=14	Balansirana F-mera	F=0.845

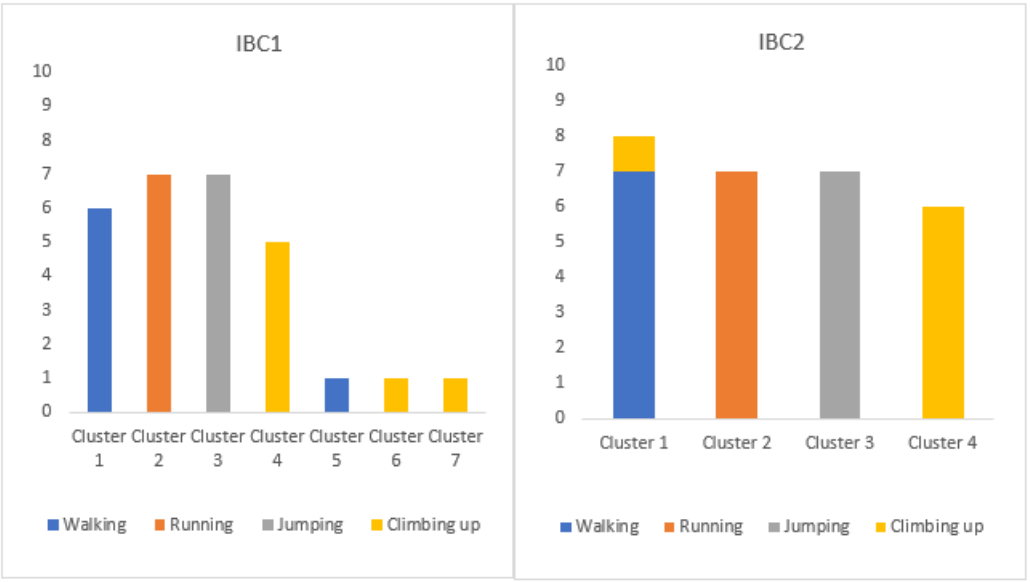
Tabela 6: Evaluacija rezultata klasterovanja dobijenih pomoću IBC2 algoritma koristeći CMU-MMAC bazu podataka

Tačnost formiranih klastera		$T = 95.65\%$	
Tačno pozitivni	TP=50	Rand indeks	RI=0.960
Tačno negativni	TN=193	Preciznost	P=0.909
Lažno pozitivni	FP=5	Odziv	R=0.909
Lažno negativni	FN=5	Balansirana F-mera	F=0.909

Dalje, predstavljeni su rezultati dobijeni pomoću RealWorld baze podataka. Skup subjekata, od kojih je svaki snimljen dok izvršava četiri različite aktivnosti (hodanje, penjanje uz stepenice, skakanje i trčanje), je podeljen na dva disjunktna podskupa. Prvi podskup je S1, S2, S3, S4, S5, S6, S7, S8, a drugi podskup je S9, S10, S11, S12, S13, S14, S15. Prvi podskup, koji sadrži 31 subjekat-aktivnost par (nedostaju podaci za jednu aktivnost za subjekat S3), korišćen je za treniranje. Drugi podskup, koji sadrži 28 subjekat-aktivnost parova, korišćen je za testiranje. Primenjujući kriterijum tačnost formiranih klastera, dobijen je

parametar $p = 15$. Dva testa su izvršena koristeći isti uzorak. Prvi, koristći IBC1 algoritam bez pretpostavke o broju klastera, a drugi koristeći IBC2 algoritam gde je broj klastera unapred dat i iznosi 4.

Rezultati klasterovanja su predstavljeni na Slici 1



Slika 1: Rezultati klasterovanja dobijeni pomoću IBC1 i IBC2 algoritama koristeći RealWorld bazu podataka

Za obuhvatniju evaluaciju, Rand indeks, preciznost, odziv i balansirana F-mera prikazane su u Tabeli 7 za slučaj bez pretpostavke o broju klastera, i u Tabeli 8 kada je broj klastera unapred dat.

Tabela 7: Evaluacija rezultata klasterovanja dobijenih pomoću IBC1 algoritma koristeći RealWorld bazu podataka

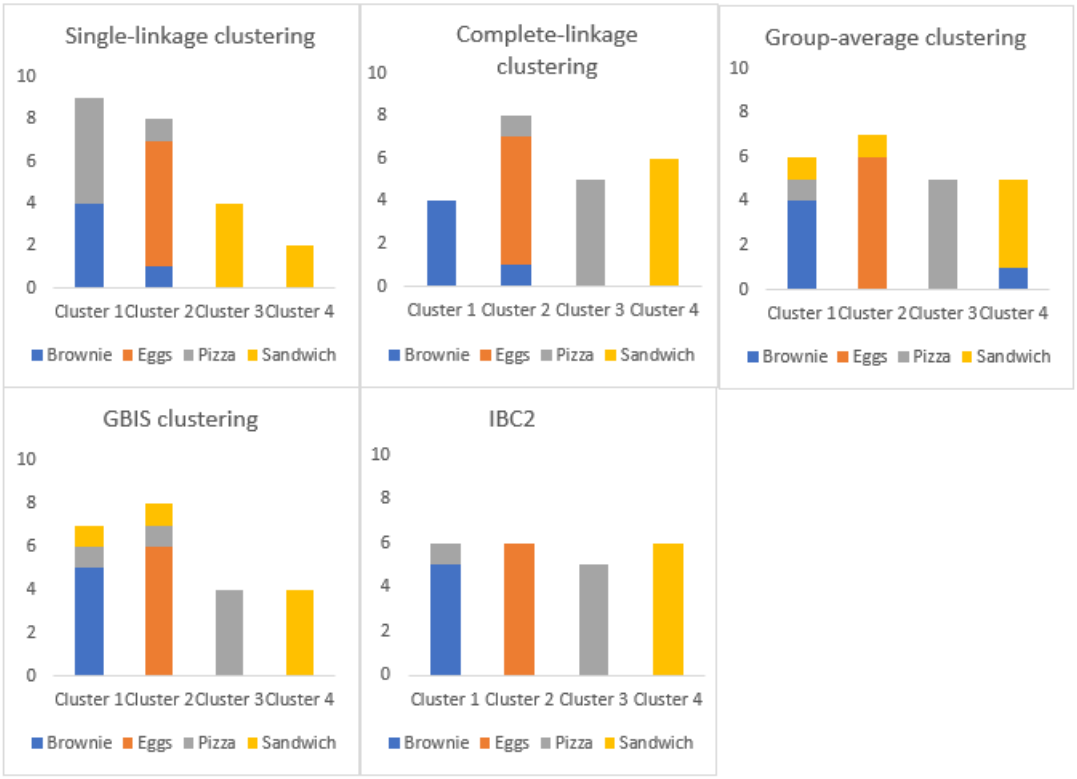
Tačnost formiranih klastera		$\boldsymbol{T = 89.29\%}$	
Tačno pozitivni	$TP = 67$	Rand indeks	$RI = 0.955$
Tačno negativni	$TN = 294$	Preciznost	$P = 1$
Lažno pozitivni	$FP = 0$	Odziv	$R = 0.798$
Lažno negativni	$FN = 17$	Balansirana F-mera	$F = 0.888$

Tabela 8: Evaluacija rezultata klasterovanja dobijenih pomoću IBC2 algoritma koristeći RealWorld bazu podataka

Tačnost formiranih klastera		$\boldsymbol{T = 96.43\%}$	
Tačno pozitivni	$TP = 78$	Rand indeks	$RI = 0.966$
Tačno negativni	$TN = 287$	Preciznost	$P = 0.918$
Lažno pozitivni	$FP = 7$	Odziv	$R = 0.929$
Lažno negativni	$FN = 6$	Balansirana F-mera	$F = 0.923$

7.3 Komparativni rezultati

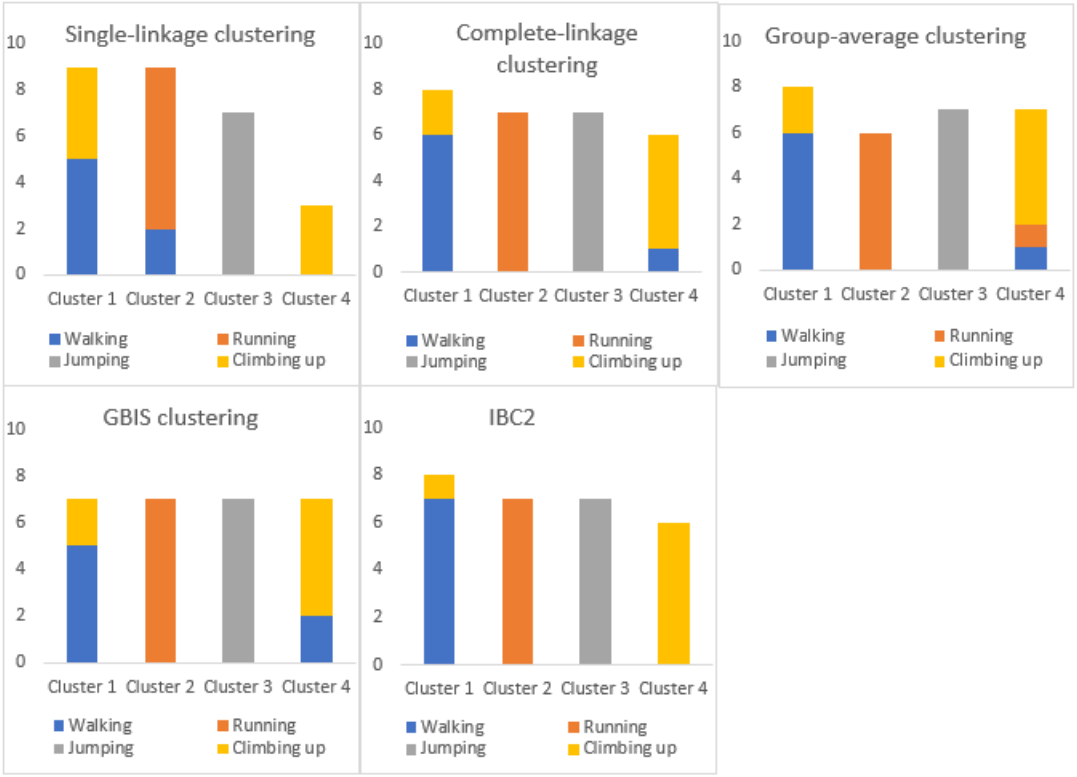
Da bi smo ocenili performanse novog algoritma, sledeći algoritmi bazirani na grafovima: jednostrukog povezivanja, potpunog povezivanja, prosečnog povezivanja [119] i segmentacije slika regionom [70] primenjeni su na dobijene grafove za slučaj kada je broj klastera unapred dat. Uporedni rezultati klasterovanja prikazani su na Slici 2 za bazu CMU-MMAC, odnosno na Slici 3 za bazu RealWorld. Uporedni rezultati evaluacije klasterovanja dati su u Tabeli 9 za bazu CMU-MMAC, odnosno u Tabeli 10 za bazu RealWorld.



Slika 2: Komparativni rezultati klasterovanja, kada je broj klastera unapred dat, koristeći CMU-MMAC bazu podataka

Tabela 9: Komparacija vrednosti parametara evaluacije dobijenih koristeći CMU-MMAC bazu podataka

	T	***RI***	***P***	***R***	***F***
Single-linkage clustering algorithm	65.22%	0.802	0.535	0.691	0.603
Complete-linkage clustering algorithm	91.30%	0.913	0.780	0.836	0.807
Group-average clustering algorithm	82.61%	0.854	0.661	0.673	0.667
GBIS clustering algorithm	82.61%	0.834	0.607	0.673	0.638
IBC2 algorithm	95.65%	0.960	0.909	0.909	0.909



Slika 3: Komparativni rezultati klasterovanja, kada je broj klastera unapred dat, koristeći RealWorld bazu podataka

Tabela 10: Komparacija vrednosti parametara evaluacije dobijenih koristeći RealWorld bazu podataka

	T	***RI***	***P***	***R***	***F***
Single-linkage clustering algorithm	67.86%	0.852	0.646	0.738	0.689
Complete-linkage clustering algorithm	89.29%	0.913	0.800	0.810	0.805
Group-average clustering algorithm	85.71%	0.881	0.729	0.738	0.733
GBIS clustering algorithm	85.71%	0.899	0.762	0.762	0.762
IBC2 algorithm	96.43%	0.966	0.918	0.929	0.923

7.4 Diskusija

Algoritmi za automatsko prepoznavanje ljudskih aktivnosti su uglavnom namenjeni za analizu tipova aktivnosti koje su relativno jednostavne i značajno se razlikuju među sobom, uključujući položaj tela, fundamentalne radnje (npr. sedenje, stajanje, hodanje, trčanje, plivanje, oblačenje, penjanje uz i silaženje niz stepenice, tapšanje, mahanje itd.) [96]. Nasuprot tome, predloženi algoritma omogućava prepoznavanje zadatih kompleksnih radnji koje nisu precizno definisane. Konkretno, različiti ljudski subjekti izvode iste tipove aktivnosti na različite načine (npr. pripremanje jela bez preciznog recepta), odnosno, pojedinačni segmenti složenih radnji mogu biti prilično različiti ili se izvršavati različitim redosledom. Dodatno, različiti tipovi kompleksnih radnji mogu imati iste ili slične segmente. Prezentovani pristup omogućava uspešno klasterovanje takvih složenih ljudskih aktivnosti.

Prvo smo analizirali performanse novog IBC1 algoritma. U Tabeli 3 su predstavljeni rezultati klasterovanja složenih ljudskih aktivnosti (CMU-MMAC baza podataka). Diskutujući ove rezultate, može se videti da su 23 subjekat-aktivnost para grupisana u šet klastera, naspram četiri data klastera. Sva četiri tražena klastera su formirana i sadrže 20 od 23 subjekat-aktivnost para, gde je tačnost formiranih klastera $T = 86.96\%$ sa samo jednim lažno pozitivnim. Zadovoljavajući rezultati klasterovanja su potvrđeni evaluacionim parametrima čije su vrednosti prikazane u Tabeli 5: Rand indeks $RI = 0.941$, preciznost $P = 0.976$, odziv $R = 0.745$ i balansirana F-mera $F = 0.845$. Na Slici 1 su prikazani rezultati klasterovanja fundamentalnih ljudskih aktivnosti. Dvadeset osam subjekat-aktivnost parova su grupisani u sedam klastera, što se može učiniti kao značajno više nego 4 prava klastera. Međutim, 25 od 28 parova su ispravno klasterisani u četiri klastera bez lažno pozitivnih. Sva četiri tražena klastera su formirana sa tačnošću formiranih klastera $T = 89.29\%$. Zadovoljavajući rezultati klasterovanja su potvrđeni evaluacionim parametrima čije su vrednosti date u Tabeli 7: Rand indeks $RI = 0.955$, preciznost $P = 1$, odziv $R = 0.798$ i balansirana F-mera $F = 0.888$. Performanse predloženog algoritma su zadovoljavajuće

na obe baze podataka. Značaj ovog algoritma potvrđen je, pre svega, dobrim rezultatima klasterovanja kompleksnih ljudskih aktivnosti.

Dalje, diskutujemo performanse IBC2 algoritma. Radi temeljnije evaluacije, na istim uzorcima testirani su sledeći algoritmi bazirani na grafovima: jednostrukog povezivanja, potpunog povezivanja, prosečnog povezivanja i segmentacije slika regionom, za slučaj kada je broj klastera unapred dat. Komparativni rezultati u slučaju složenih ljudskih aktivnosti (CMU-MMAC baza podataka) su prikazani na Slici 2. Može se zaključiti da su primenom svih algoritama formirana sva četiri klastera, osim u slučaju algoritma jednostrukog povezivanja. Komparativni rezultati evaluacije su prikazani u Tabeli 9. Najbolji rezultati evaluacije po svim parametrima su dobijeni za IBC2 algoritam: tačnost formiranih klastera $T = 95.65\%$, Rand indeks $RI = 0.960$, preciznost $P = 0.909$, odziv $R = 0.909$ i balansirana-F mera $F = 0.909$. Komparativni rezultati, u slučaju fundamentalnih ljudskih aktivnosti (RealWorld baza podataka), su ilustrovani na Slici 3. Mozemo zaključiti da su formirana sva četiri tražena klastera primenom svih algoritama, izuzev algoritma jednostrukog povezivanja koji ima tendenciju spajanja klastera. Najbolji rezultati evaluacije po svim parametrima su dobijeni za IBC2 algoritam. Tačnost formiranih klastera $T = 96.43\%$, Rand indeks $RI = 0.966$, preciznost $P = 0.918$, odziv $R = 0.929$ i balansirana-F mera $F = 0.923$.

Da bismo kompletirali komparativnu analizu, dobijeni rezultati su upoređeni sa rezultatima dobijenim pomoću dve različite metode korišćene za klasterovanje ljudskih aktivnosti. Prema preglednom radu [15], dve najčešće korišćene metode su k-sredina (eng. k-means) i pod-klasterovanje (eng. Sub-clustering). U istom preglednom radu predstavljeni su najznačajniji rezultati, za svaku od korišćenih baza podataka, pomoću gore navedenih metoda. Najbolji rezultati objavljeni u časopisima predstavljeni su u Tabeli 11.

Tabela 11: Upoređivanje metoda

Metoda	**Baza podataka**	***RI***
K-means	VanKasteren	0.872
	WISDM	0.710
	Liara	0.860
	Opportunity	0.868
	MHealth	0.786
	UCI HAR	0.794
Sub-clustering	VanKasteren	0.894
	Casas Aruba	0.898
	Casas Kyoto	0.891
IBC2	RealWorld	0.966

Predloženi metod sa IBC2 algoritmom daje najbolje rezultate klasterovanja ocenjene pomoću Rand indeksa.

Većina algoritama za klasterovanje je bazirana isključivo na distancama između para tačaka, ili na povezanosti. Predloženi IBC1 algoritma kombinuje ova dva principa. Štaviše, metrika nije potrebna. Glavna karakteristika ovog algoritma je sposobnost da analizira grafove sa klasterima koji se preklapaju, kao i sa klasterima sa značajnim međuklasterskim sličnostima. Nedostatak ovog algoritma je visoka vremenska kompleksnost za veliki broj tačaka podataka. IBC2 algoritam je neefikasan kada je broj tačaka po klasterima neujednačen.

8 Zaključak

U ovoj disertaciji, predložen je novi pristup koji omogućava unapređenje klasterovanja ljudskih aktivnosti u realnom okruženju. U tu svrhu konstruisana su dva nova algoritma klasterovanja, bazirana na grafovima; u slučaju kada broj klastera nije poznat i u slučaju kada je broj klastera unapred dat. Takođe, predstavljena je metoda simboličkog modeliranja prostorno-vremenskih signala dobijenih sa prenosivih inercijalnih mernih uređaja, kao i adekvatna mera sličnosti podataka dobijenih modeliranjem ljudskih aktivnosti.

Predložen pristup je testiran na dve baze podataka, CMU-MMUC koja sadrži zabeleške složenih ljudskih aktivnosti i RealWorld bazom, koja sadrži zabeleške fundamentalnih ljudskih aktivnosti. Takođe je izvršena detaljna komparativna analiza predloženog pristupa sa dobro poznatim algoritmima na istim uzorcima.

Za potpuniju validaciju performansi algoritama klasterovanja, predložena je nova mera evaluacije - tačnost formiranih klastera. U slučaju oba eksperimenta, predložen pristup je značajno unapredio rezultate do kojih se došlo primenom drugih algoritama u rešavanju problema klasterovanja ljudskih aktivnosti.

Naučni doprinos ove doktorske disertacije je potvrđen publikovanjem u međunarodnom časopisu izuzetnih vrednosti [48]. U disertaciji su detaljno izloženi rezultati objavljeni u međunarodnim časopisima.

Uzevši u obzir sve što je prethodno izloženo, možemo izdvojiti glavne naučne doprinose ove disertacije:

- Novi algoritam za klasterovanje težinskih grafova, pogodan za klasterovanje složenih ljudskih aktivnosti, kada broj klastera nije unapred poznat.
- Novi algoritam za klasterovanje težinskih grafova, pogodan za klasterovanje složenih ljudskih aktivnosti, kada je broj klastera unapred dat.
- Unapređenje tehnike obrade signala sa prenosivih uređaja za potrebe analize ljudskih aktivnosti.
- Nova metoda evaluacije rezultata klasterovanja.

Na osnovu prikazanih rezultata istraživanja, možemo zaključiti da su potvrđene opšta i posebne hipoteze, koje su poslužile kao polazna osnova za istraživanje i eksperimente.

Uz razvoj kompjuterskog hardvera i činjenicu da postaje pristupačniji, kompaktniji i brži, postaje sve prisutniji u svakodnevnom životu. To dovodi do velikog porasta korišćenja nosivih uređaja opremljenih sa IMU (pametni telefoni, pametni satovi i slični uređaji) samim tim i povećanje korišćenja IMU podataka za analizu i prepoznavanje ljudskih aktivnosti. Buduća istraživanja mogu uključiti korišćenje novih karakteristika izvedenih iz originalnih (npr. brzina i relativni položaj mogu biti izvedeni iz podataka o trenutnom ubrzanju) ili analizu više senzora postavljenih na različite delove tela subjekata.

9 Literatura

[1] Abdel-Basset, M., Abdel-Fatah, L., Sangaiah, A. K., Metaheuristic algorithms: A comprehensive review, Computational intelligence for multimedia big data on the cloud with engineering applications, pp. 185.231, 2018.

[2] Adabonyan I, Loustalot F, Kruger J, Carlson SA, Fulton JE. Prevalence of highly active adults-Behavioral risk factor surveillance system, 2007. Prev Med 2010; 51(2): 139-143.

[3] Aguiar P, Neto D, Lambaz R, Chick J, Ferrinho P. Prognostic factors during outpatient treatment for alcohol dependence: cohort study with 6 months of treatment follow-up. Alcohol Alcoholism 2012; 47: 702-10.

[4] Ahirwar R. A Novel K means clustering algorithm for large datasets based on divide and conquer technique. Int J Comput Sci Inf Techn 2014; 5 (1):301-305.

[5] Ahmad A, Dey L. A k-means type clustering algorithm for mixed numeric and categorical datasets. Data Knowl Eng 2007; 63 (2): 503-527.

[6] Ahmad A, Dey L. A k-means type clustering algorithm for subspace clustering of mixes numeric and categorical datasets. Pattern Recognit Lett 2011;32: 1062- 1069.

[7] Akaike H. A new look at the statistical model identification. IEEE Trans Autom Control. 1974, 19(6): 716-723.

[8] Alam AY, Iqbal A, Mohamud KB, Laporte RE, Ahmed A, Nishtar S. Investigating socio-economic-demographic determinants of tobacco use in Rawalpindi, Pakistan. BMC Public Health 2008; 8(1): 50.

[9] Aldana SG, Greenlaw RL, Diehl HA, Salberg A, Merrill RM, Ohmine S, Thomas C. Effects of an intensive diet and physical activity modification program on the health risks of adults. J Amer Diet Ass 2005; 105:371-381.

[10] Aldous, D. and Fill, J. (in preparation). Reversible Markov Chains and Random Walks on Graphs. online version available at http://www.stat.berkeley.edu/users/aldous/RWG/book.html.

[11] Al-Salami, N. M., Evolutionary algorithm definition, American Journal of Engineering and Applied Sciences, vol. 2, no. 4, pp. 789.795, 2009.

[12] Al-Sultana KS, Khan MM. Computational experience on four algorithms fr the hard clustering problem. Pattern Recognit Lett 1976; 17(3): 295-308.

[13] Anderberg MR. Cluster Analysis for Applications. New York: Academic, 1973

[14] Anderson, J. A., Discrete mathematics with combinatorics. Prentice Hall, 2001.

[15] Ariza Colpas, P., Vicario, E., De-La-Hoz-Franco, E., Pineres-Melo, M., Oviedo-Carrascal, A., Patara, F., Unsupervised Human Activity Recognition Using the Clustering Approach: A Review. Sensors 2020, 20, 2702.

[16] Arthur, D., Vassilvitskii, S., k-means++: The Advantages of Careful Seeding, In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, str. 1027-1035, 2007.

[17] Audibert, M., von Luxburg, U., (2007). Graph laplacians and their convergence on random neighborhood graphs. JMLR, 8, 1325-1370.

[18] Avci, A., Bosch, S., Marin-Perianu, M., Marin-Perianu, R., Havinga, P. (2010) Activity Recognition Using Inertial Sensing for Healthcare, Wellbeing and Sports Applications: A Survey, 23rd International Conference on Architecture of Computing Systems 2010, Hannover, Germany, pp. 1-10.

[19] Bacanin, N., Stoean, R.,Zivkovic, M., Petrovic, A., Rashid, T. A., Bezdan, T., Performance of a Novel Chaotic Firefly Algorithm with Enhanced Exploration for Tackling Global Optimization Problems: Application for Dropout Regularization, Mathematics, Vol. 9, No. 21, pp. 1 - 33, Oct, 2021.

[20] Bacanin, N., Bezdan, T., Venkatachalam, K., Al-Turjman, F., Optimized convolutional neural network by firefly algorithm for magnetic resonance image classification of glioma brain tumor grade, Journal of Real-Time Image Processing, pp. 1 - 14, Apr, 2021.

[21] Bacanin, N., Bezdan, T., Tuba, E., Strumberger, I., Tuba, M., Monarch Butterfly Optimization Based Convolutional Neural Network Design, Mathematics, Vol. 8, No. 6, pp. 936 - 936, Jun, 2020.

[22] Bacanin, N., Bezdan, T., Tuba, E., Strumberger, I., Tuba, M., Optimizing Convolutional Neural Network Hyperparameters by Enhanced Swarm Intelligence Metaheuristics, Algorithms, Vol. 13, No. 6, pp. 67:1 - 67:33, Mar, 2020.

[23] Badawi, A., Al-Kabbany, A., Shaban, H. (2020) Sensor Type, Axis, and Position-Based Fusion and Feature Selection for Multimodal Human Daily Activity Recognition in Wearable Body Sensor Networks, Journal of Healthcare Engineering, Volume 2020, Article ID 7914649

[24] Bach, F. and Jordan, M. (2004). Learning spectral clustering. In S. Thrun, L. Saul, and B. Scholkopf (Eds.), Advances in Neural Information Processing Systems 16 (NIPS) (pp. 305-312). Cambridge, MA: MIT Press.

[25] Bapat, R., Gutman, I., and Xiao, W. (2003). A simple method for computing resistance distance. Z. Naturforsch., 58, 494 - 498.

[26] Barnard, S., Pothen, A., and Simon, H. (1995). A spectral algorithm for envelope reduction of sparse matrices. Numerical Linear Algebra with Applications, 2 (4), 317-334.

[27] Bar-Yam, Y., General features of complex systems, Encyclopedia of Life Support Systems, 2002.

[28] Basha, J., Bacanin, N., Vukobrat, N., Zivkovic, M., Venkatachalam, K., Hubálovský, S., Trojovský, P., Chaotic Harris Hawks Optimization with Quasi-Reflection-Based Learning: An Application to Enhance CNN Design, SENSORS, Vol. 21, No. 19, pp. 1 - 33, Oct, 2021.

[29] Belhaouari, S., Ahmed, S., Mansour, S., Optimized K-Means Algorithm, Mathematical Problem in Engineering, Vol. 2014, pp. 1-14, September 2014.

[30] Belkin, M. (2003). Problems of Learning on Manifolds. PhD Thesis, University of Chicago.

[31] Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 15 (6), 1373-1396.

[32] Belkin, M. and Niyogi, P. (2005). Towards a theoretical foundation for Laplacian-based manifold methods. In P. Auer and R. Meir (Eds.), Proceedings of the 18th Annual Conference on Learning Theory (COLT) (pp. 486-500). Springer, New York.

[33] Bellman, R., On a routing problem, Quarterly of Applied Mathematics, vol. 16, no. 1, pp. 87.90, 1958.

[34] Ben-David, S., von Luxburg, U., and Pal, D. (2006). A sober look on clustering stability. In G. Lugosi and H. Simon (Eds.), Proceedings of the 19th Annual Conference on Learning Theory (COLT) (pp. 5-19). Springer, Berlin.

[35] Bengio, Y., Delalleau, O., Roux, N., Paiement, J., Vincent, P., and Ouimet, M. (2004). Learning eigenfunctions links spectral embedding and kernel PCA. Neural Computation, 16, 2197-2219.

[36] Ben-Hur, A., Elisseeff, A., and Guyon, I. (2002). A stability based method for discovering structure in clustered data. In Pacific Symposium on Biocomputing (pp. 6-17).

[37] Bezdan, T., Stoean, C., Al Naamany, A., Bacanin, N., Rashid, T. A., Zivkovic, M., Venkatachalam, K., Hybrid Fruit-Fly Optimization Algorithm with K-Means for Text Document Clustering, Mathematics, Vol. 9, No. 16, pp. 1 - 19, Aug, 2021

[38] Bhatia, R. (1997). Matrix Analysis. Springer, New York.

[39] Bie, T. D. and Cristianini, N. (2006). Fast SDP relaxations of graph cut clustering, transduction, and other combinatorial problems . JMLR, 7, 1409-1436.

[40] Biggs, N., Lloyd, E. K., Wilson, R. J., Graph Theory, 1736.1936. Oxford University Press, 1986.

[41] Bock, H., Probabilistic aspects in cluster analysis, Springer-Verlag, Augsburg, 1989.

[42] Bolla, M. (1991). Relations between spectral and classification properties of multigraphs (Technical Report No. DIMACS-91-27). Center for Discrete Mathematics and Theoretical Computer Science.

[43] Borg I., Groenen P.J.F., Mair P. (2013), Applied Multidimensional Scaling, Springer

[44] Borg I., Groenen P. J. F. (2005), Modern multidimensional scaling, New York: Springer.

[45] Bremaud, P. (1999). Markov chains: Gibbs fields, Monte Carlo simulation, and queues. New York: Springer-Verlag.

[46] Brito, M., Chavez, E., Quiroz, A., and Yukich, J. (1997). Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection. Statistics and Probability Letters, 35, 33-42.

[47] Bui, T. N. and Jones, C. (1992). Finding good approximate vertex and edge partitions is NP-hard. Inf. Process. Lett., 42 (3), 153-159.

[48] Budimirovic, N., Bacanin, N., Novel Algorithms for Graph Clustering Applied to Human Activities. Mathematics 2021, Volume 9, Issue 10, 1089. https://doi.org/10.3390/math9101089

[49] Chapelle, O., Scholkopf, B., and Zien, A. (Eds.). (2006). Semi-Supervised Learning. MIT Press, Cambridge.

[50] Chau, M., Cheng, R., Kao, B., Ng, J., Uncertain data mining: An example in clustering location data, In PAKDD Singapore 2006, pp. 199-204, 2006.

[51] Chung, F. (1997). Spectral graph theory (Vol. 92 of the CBMS Regional Conference Series in Mathematics). Conference Board of the Mathematical Sciences, Washington.

[52] Clustering basic benchmark, University of Eastern Finland, dostupno na: http://cs.joensuu.fi/sipu/datasets/ [23.8.2020.]

[53] Cvetković D., Simić S.K., Graph spectra in computer science, Linear Algebra Appl., 434(2011), 1545-1562.

[54] Cvetković D., A graph theoretical procedure for clustering binary vectors, Ars Combinatoria, 46(1997), 267-276. MR 98d: 05096, Zbl. 932, 68064.

[55] Cvetković, D., Milić M., Teorija grafova i njene primene. Beogradski izdavačko-grafički zavod, 1971.

[56] De la Torre, F., Hodgins, J., Montano, J., Valcarcel, S., Forcada, R., Macey, J. (2009) Guide to the Carnegie Mellon University Multimodal Activity (CMU-MMAC) Database, Tech. report CMU-RI-TR-08-22, Robotics Institute, Carnegie Mellon University.

[57] Dhillon, I. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In Proceedings of the seventh ACM SIGKDD in-

ternational conference on Knowledge discovery and data mining (KDD) (pp. 269-274). New York: ACM Press.

[58] Dhillon, I., Guan, Y., and Kulis, B. (2005). A unified view of kernel k-means, spectral clustering, and graph partitioning (Technical Report No. UTCS TR-04-25). University of Texas at Austin.

[59] Dijkstra E. W. et al., A note on two problems in connexion with graphs, Numerische Mathematik, vol. 1, no. 1, pp. 269.271, 1959.

[60] Ding, C., He, X., Zha, H., Gu, M., and Simon, H. (2001). A min-max cut algorithm for graph partitioning and data clustering. In Proceedings of the first IEEE International Conference on Data Mining (ICDM) (pp. 107-114). Washington, DC, USA: IEEE Computer Society.

[61] Donath, W. E. and Hoffman, A. J. (1973). Lower bounds for the partitioning of graphs. IBM J. Res. Develop., 17, 420-425.

[62] Dorigo, M., Birattari, M., Stutzle, T., Ant colony optimization, IEEE Computational Intelligence Magazine, vol. 1, no. 4, pp. 28.39, 2006.

[63] Eberhart R. C., Shi, Y., Computational intelligence: concepts to implementations. Elsevier, 2011.

[64] Eiben, A. E., Smith, J. E., Introduction to evolutionary computing. Springer, 2015.

[65] Euler, L., Solutio problematis ad geometriam situs pertinentis, Commentarii Academiae Scientiarum Petropolitanae, pp. 128.140, 1741.

[66] Everitt B., Hothorn T. (2012), An Introduction to Applied Multivariate Analysis with R, Springer

[67] Everitt B., Rabe-Hesketh S. (1997), The Analysis of Proximity Data. London: Chapman and Hall/CRC

[68] Everit B. (2005), An R and S-Plus Companion to Multivariate Analysis, Springer

[69] Everitt B., Landau S., Leese M. (2001), Cluster Analysis, Fourth edition, Arnold.

[70] Felzenszwalb, P.F., Huttenlocher, D.P. (2004) Efficient Graph-Based Image Segmentation, International Journal of Computer Vision, 59, pp. 167-181.

[71] Feo T. A., Resende, M. G., Greedy randomized adaptive search procedures, Journal of Global Optimization, vol. 6, no. 2, pp. 109.133, 1995.

[72] Fiedler, M. (1973). Algebraic connectivity of graphs. Czechoslovak Math. J., 23, 298-305.

[73] Fouss, F., Pirotte, A., Renders, J.-M., and Saerens, M. (2007). Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Trans. Knowl. Data Eng, 19 (3), 355-369.

[74] Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. JASA, 97, 611-631.

[75] Gan, G., Ma, C., Wu, J., Data Clustering: Theory, Algorithms, and Applications, ASA-SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia, ASA, Alexandria, VA, 2007.

[76] Gill, P. E., Murray, W., Saunders, M. A., Tomlin, J. A., Wright,M. H., George B. Dantzig and systems optimization, Discrete Optimization, vol. 5, no. 2, pp. 151.158, 2008.

[77] Gine, E. and Koltchinskii, V. (2005). Empirical graph Laplacian approximation of Laplace-Beltrami operators: large sample results. In Proceedings of the 4th International Conference on High Dimensional Probability (pp. 238-259).

[78] Glover, F., Laguna, M., Marti, R., Scatter search, Advances in Evolutionary Computing, pp. 519.537, 2003.

[79] Glover, F., Tabu search . part I, ORSA Journal on Computing, vol. 1, no. 3, pp. 190.206, 1989.

[80] Glover, F., Tabu search . part II, ORSA Journal on Computing, vol. 2, no. 1, pp. 4.32, 1990.

[81] Gnjatović, M., Nikolić, V., Joksimović, D., Maček, N., Budimirović, N. (2020) An Approach to Human Activity Clustering using Inertial Measurement Data, X International Scientific Conference Archibald Reiss, Republic of Serbia, Belgrade, 18-19 November 2020

[82] Golub, G. and Van Loan, C. (1996). Matrix computations. Baltimore: Johns Hopkins University Press.

[83] Guattery, S. and Miller, G. (1998). On the quality of spectral separators. SIAM Journal of Matrix Anal. Appl., 19 (3), 701-719.

[84] Gutman, I. and Xiao, W. (2004). Generalized inverse of the Laplacian matrix and some applications. Bulletin de l Academie Serbe des Sciences at des Arts (Cl. Math. Natur.), 129, 15-23.

[85] Hagen, L. and Kahng, A. (1992). New spectral methods for ratio cut partitioning and clustering. IEEE Trans. Computer-Aided Design, 11 (9), 1074-1085.

[86] Hastie, T., Tibshirani, R., and Friedman, J. (2001). The elements of statistical learning. New York: Springer.

[87] Hein, M. (2006). Uniform convergence of adaptive graph-based regularization. In Proceedings of the 19th Annual Conference on Learning Theory (COLT) (pp. 50-64). Springer, New York.

[88] Hein, M., Audibert, J.-Y., and von Luxburg, U. (2005). From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. In P. Auer and R. Meir (Eds.), Proceedings of the 18th Annual Conference on Learning Theory (COLT) (pp. 470-485). Springer, New York.

[89] Hendrickson, B. and Leland, R. (1995). An improved spectral graph partitioning algorithm for mapping parallel computations. SIAM J. on Scientific Computing, 16, 452-469.

[90] Huang X., Lai W., Clustering graphs for visualiyation via node similarities, Journal of Visual Languages and Computing, vol.17, issue 3, 2006, 225-253.

[91] Human Activity Recognition. Available online: https://sensor.informatik.uni-mannheim.de/ (accessed on 20 March 2021).

[92] Hussain, Z., Sheng, M., Zhang, W.E. (2019) Different Approaches for Human Activity Recognition: A Survey, arXiv, 1906.05074, Downloaded July 16th 2020, https://arxiv.org/abs/1906.05074.

[93] Izenman A. J. (2008), Modern Multivariate Techniques, New York: Springer-Verlag

[94] Janičić, P., Matematička logika u računarstvu. Matematički fakultet, Beograd, 2008.

[95] Joachims, T. (2003). Transductive Learning via Spectral Graph Partitioning. In T. Fawcett and N. Mishra (Eds.), Proceedings of the 20th international conference on machine learning (ICML) (pp. 290-297). AAAI Press.

[96] Jobanputra, C., Bavishi, J., Doshi, N. (2019) Human Activity Recognition: A Survey, Procedia Computer Science, 15, pp. 698-703.

[97] Jurafsky, D., Martin, J.H. (2009) Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics, 2nd edition, Prentice-Hall.

[98] Kannan S., Ramathilagam S., Chung P., Effective fuzzy c-means clustering algorithms for data clustering problems, Expert Systems with Applications, vol.39, issue 7, 2012 6292-6300.

[99] Kannan, R., Vempala, S., and Vetta, A. (2004). On clusterings: Good, bad and spectral. Journal of the ACM, 51 (3), 497-515.

[100] Karmarkar, N., A new polynomial-time algorithm for linear programming, Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, pp. 302.311, 1984.

[101] Karp, R. M., Reducibility among combinatorial problems, Complexity of Computer Computations, pp. 85.103, 1972.

[102] Keim, D., Hinneburg, A., Optimal grid-clustering: Towards breaking the curse of dimensionality in high-dimensional clustering. In Proceedings of the 25th international conference on very large data bases (VLDB 99), Morgan Kaufmann, San Francisco, CA, 1999.

[103] Kempe, D. and McSherry, F. (2004). A decentralized algorithm for spectral analysis. In Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC) (pp. 561-568). New York, NY, USA: ACM Press.

[104] Kennedy, J., Swarm intelligence, Handbook of Nature-Inspired and Innovative Computing, pp. 187.219, 2006.

[105] Khachiyan, L. G., A polynomial algorithm in linear programming, Doklady Akademii Nauk, vol. 244, no. 5, pp. 1093.1096, 1979.

[106] Klein, D. and Randic, M. (1993). Resistance distance. Journal of Mathematical Chemistry, 12, 81-95.

[107] Konig, D., Theorie der endlichen und unendlichen Graphen: Kombinatorische Topologie der Streckenkomplexe, vol. 16. Akademische Verlagsgesellschaft mbh, 1936.

[108] Koren, Y. (2005). Drawing graphs by eigenvectors: theory and practice. Computers and Mathematics with Applications, 49, 1867-1888.

[109] Krzanowski, W. J. (1988), Principles of Multivariate Analysis, Oxford, UK: Oxford University Press

[110] Kruskal J., Wish M. (1977), Multidimensional Scaling

[111] Lafon, S. (2004). Diffusion maps and geometric harmonics. PhD Thesis, Yale University.

[112] Lang, K. (2006). Fixing two weaknesses of the spectral method. In Y. Weiss, B. Scholkopf, and J. Platt (Eds.), Advances in Neural Information Processing Systems 18 (pp. 715-722). Cambridge, MA: MIT Press.

[113] Lange, T., Roth, V., Braun, M., and Buhmann, J. (2004). Stability-based validation of clustering solutions. Neural Computation, 16 (6), 1299-1323.

[114] Levenshtein, V.I. (1966) Binary codes capable of correcting deletions, insertions, and reversals, Cybernetics and Control Theory, 10(8), pp. 707-710 (Original in Doklady Akademii Nauk SSSR 163(4): 845-848, 1965)

[115] Lozanov-Crvenkovic Z. (2011), Statistika, Novi Sad

[116] Lovasz, L. (1993). Random walks on graphs: a survey. In Combinatorics, Paul Erdos is eighty (pp. 353-397). Budapest: Janos Bolyai Math. Soc.

[117] Lutkepohl, H. (1997). Handbook of Matrices. Chichester: Wiley.

[118] Mangasarian, O. L., Nonlinear programming. SIAM, 1994.

[119] Manning C., Raghavan P. and Schutze H., Introduction to Information Retrieval, Cambridge University Press. 2008., chapter 17

[120] Manson, S. M., Simplifying complexity: a review of complexity theory, Geoforum, vol. 32, no. 3, pp. 405.414, 2001.

[121] Marić, M., Stanimirović Z., Božović, S., Hybrid metaheuristic method for determining locations for long-term health care facilities, Annals of Operations Research, vol. 227, no. 1, pp. 3.23, 2015.

[122] Marić, M., Stanimirović Z., and P. Stanojević, An efficient memetic algorithm for the uncapacitated single allocation hub location problem, Soft Computing, vol. 17, no. 3, pp. 445.466, 2013.

[123] Marić, M., Stanimirović Z., Djenić, A., Stanojević, P., Memetic algorithm for solving the multilevel uncapacitated facility location problem, Informatica, vol. 25, no. 3, pp. 439.466, 2014.

[124] Marti, R., Laguna, M., Glover, F., Principles of scatter search, European Journal of Operational Research, vol. 169, no. 2, pp. 359-372, 2006.

[125] Meila, M. and Shi, J. (2001). A random walks view of spectral segmentation. In 8th International Workshop on Artificial Intelligence and Statistics (AISTATS).

[126] Mohar, B. (1991). The Laplacian spectrum of graphs. In Graph theory, combinatorics, and applications. Vol. 2 (Kalamazoo, MI, 1988) (pp. 871-898). New York: Wiley.

[127] Mohar, B. (1997). Some applications of Laplace eigenvalues of graphs. In G. Hahn and G. Sabidussi (Eds.), Graph Symmetry: Algebraic Methods and Applications (Vol. NATO ASI Ser. C 497, pp. 225-275). Kluwer.

[128] Morrison, D. R., Jacobson, S. H., Sauppe, J. J., Sewell, E. C., Branchand-bound algorithms: A survey of recent advances in searching, branching, and pruning, Discrete Optimization, vol. 19, pp. 79.102, 2016.

[129] Moscato, P., On evolution, search, optimization, genetic algorithms and martial arts: towards memetic algorithms, Caltech Concurrent Computation Program, i. 826, 1989.

[130] Nadler, B., Lafon, S., Coifman, R., and Kevrekidis, I. (2006). Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators. In Y. Weiss, B. Scholkopf, and J. Platt (Eds.), Advances in Neural Information Processing Systems 18 (pp. 955-962). Cambridge, MA: MIT Press.

[131] Ng, A., Jordan, M., and Weiss, Y. (2002). On spectral clustering: analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14 (pp. 849-856). MIT Press.

[132] Norris, J. (1997). Markov Chains. Cambridge: Cambridge University Press.

[133] Ogbuabor, G., La, R. (2018) Human Activity Recognition for Healthcare using Smartphones, ICMLC 2018: Proceedings of the 2018 10th International Conference on Machine Learning and Computing

[134] Patric L. Odel and Benjamin S. Duran, (1974), Cluster Analysis: A Survey, Springer

[135] Penrose, M. (1999). A strong law for the longest edge of the minimal spanning tree. Ann. of Prob., 27 (1), 246-260.

[136] Pothen, A., Simon, H. D., and Liou, K. P. (1990). Partitioning sparse matrices with eigenvectors of graphs. SIAM Journal of Matrix Anal. Appl., 11, 430-452.

[137] Rencher A.C. (2002), Methods of Multivariate Analysis, Second edition, Wiley.

[138] Saerens, M., Fouss, F., Yen, L., and Dupont, P. (2004). The principal components analysis of a graph, and its relationships to spectral clustering. In Proceedings of the 15th European Conference on Machine Learning (ECML) (pp. 371-383). Springer, Berlin.

[139] Savelsbergh, M. W. P., Branch and price: Integer programming with column generation, Encyclopedia of Optimization, pp. 328.332, 2009.

[140] Schimke, S., Vielhauer, C., Dittmann, J. (2004) Using adapted Levenshtein distance for on-line signature authentication, Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004, Cambridge, 2004, pp. 931-934, Vol.2.

[141] Schermer, D., Moeini, M., Wendt, O., A matheuristic for the vehicle routing problem with drones and its variants, Transportation Research Part C: Emerging Technologies, vol. 106, pp. 166.204, 2019.

[142] Schubert, E., Rousseeuw, P.J., Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms, Similarity Search and Applications, Springer International Publishing, 11807, pp. 171-187

[143] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22 (8), 888-905.

[144] Simon, H. (1991). Partitioning of unstructured problems for parallel processing. Computing Systems Engineering, 2, 135-148.

[145] Spielman, D. and Teng, S. (1996). Spectral partitioning works: planar graphs and finite element meshes. In 37th Annual Symposium on Foundations of Computer Science (Burlington, VT, 1996) (pp. 96-105). Los Alamitos, CA: IEEE Comput. Soc. Press. (See also extended technical report.)

[146] Stanimirović, Z., Nelinearno programiranje. Matematički fakultet, Beograd, 2014.

[147] Stanimirović, Z., Marić, M., Radojičić, N., Bočović, S., Two efficient hybrid metaheuristic methods for solving the load balance problem, Applied and Computational Mathematics, vol. 13, no. 3, pp. 332.349, 2014.

[148] Stanojević, P., Marić, M., Stanimirović, Z., A hybridization of an evolutionary algorithm and a parallel branch and bound for solving the capacitated single allocation hub location problem, Applied Soft Computing, vol. 33, pp. 24.36, 2015.

[149] Stewart, G. and Sun, J. (1990). Matrix Perturbation Theory. New York: Academic Press.

[150] Still, S. and Bialek, W. (2004). How many clusters? an information-theoretic perspective. Neural Comput., 16 (12), 2483-2506.

[151] Stoer, M. and Wagner, F. (1997). A simple min-cut algorithm. J. ACM, 44 (4), 585-591.

[152] Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a dataset via the gap statistic. J. Royal. Statist. Soc. B, 63 (2), 411-423.

[153] Ting, T., Yang, X. S., Cheng, S., Huang, K., Hybrid metaheuristic algorithms: past, present, and future, Recent advances in swarm intelligence and evolutionary computation, pp. 71.83, 2015.

[154] Tiwari, M., Singh, R., Comparative Investigation of K-Means and K-Medoid Algorithm on Iris Data, International Journal of Engineering Research and Development, No. 4, Vol. 8, pp. 69-72, November 2012.

[155] Vanneste P., Oramas J., Verelst T., Tuytelaars T., Raes A., Depaepe F., Noortgate W., (2021), Computer Vision and Human Behaviour, Emotion and Cognition Detection: A Use Case on Student Engagement, Mathematics, Volume 9, Issue 3.

[156] Van Driessche, R. and Roose, D. (1995). An improved spectral bisection algorithm and its application to dynamic load balancing. Parallel Comput., 21 (1), 29-48.

[157] Villegas, J. G., Prins, C., Prodhon, C., Medaglia, A. L., Velasco, N., A matheuristic for the truck and trailer routing problem, European Journal of Operational Research, vol. 230, no. 2, pp. 231.244, 2013.

[158] Voloshin, V. I., Introduction to graph and hypergraph theory. Nova Science Publication, 2009.

[159] Von Luxburg, U., A tutorial on spectral clustering, Statist. Comput. 17 (2007) 395-416. Van Laarhoven P. J., Aarts, E. H., Simulated annealing, Simulated Annealing: Theory and Applications, pp. 7.15, 1987.

[160] Von Luxburg, U., Belkin, M., and Bousquet, O. (to appear). Consistency of spectral clustering. Annals of Statistics. (See also Technical Report 134, Max Planck Institute for Biological Cybernetics, 2004)

[161] Von Luxburg, U., Bousquet, O., and Belkin, M. (2004). On the convergence of spectral clustering on random samples: the normalized case. In J. Shawe-Taylor and Y. Singer (Eds.), Proceedings of the 17th Annual Conference on Learning Theory (COLT) (pp. 457-471). Springer, New York.

[162] Von Luxburg, U., Bousquet, O., and Belkin, M. (2005). Limits of spectral clustering. In L. Saul, Y. Weiss, and L. Bottou (Eds.), Advances in Neural Information Processing Systems (NIPS) 17 (pp. 857-864). Cambridge, MA: MIT Press.

[163] Wagner, D. and Wagner, F. (1993). Between min cut and graph bisection. In Proceedings of the 18th International Symposium on Mathematical Foundations of Computer Science (MFCS) (pp. 744-750). London: Springer.

[164] Wang, X., Wang, Y., Wang, L., Improving fuzzy c-means clustering based on feature-weight learning, Pattern Recognition Letters, Vol. 25, No. 10, pp. 1123-1132, July 2004.

[165] Winkler, R., Klawonn, F., Kruse, R., Fuzzy C-Means in High Dimensional Spaces, International Journal of Fuzzy System Applications, Vol. 11, June 2010.

[166] Xiao, L., Hung, E., An Efficient Distance Calculation Method for Uncertain Objects, Computational Intelligence and Data Mining, CIDM, 2007.

[167] Zhang, B., Comparison of the Performance of Center-Based Clustering Algorithms, PAKDD 2003: Advances in Knowledge Discovery and Data Mining, pp. 63-74, 2003.

10 Biografija

Nebojša Budimirović je rođen 1979. godine u Šapcu. Osnovnu školu i gimnaziju je završio u Šapcu. Dobitnik je diploma "Vuk Karadžic" i "Mihailo Petrović - Alas" za matematiku i prve nagrade na republičkom takmičenju mladih matematičara Srbije. Diplomirao je na Departmanu za matematiku, Prirodno-matematičkog fakulteta u Novom Sadu 2013. godine sa prosečnom ocenom 8,00 i stekao zvanje Diplomirani inženjer matematike. Odbranio je diplomski rad pod naslovom "Primena matematičke logike u fazi skupovima" sa ocenom 10. Radio je na Fakultetu za kompjuterske nauke Megatrend univerziteta u Beogradu u zvanju saradnika i Akademiji strukovnih studija Šabac u Šapcu u kojoj i sada radi u zvanju asistenta. Pored maternjeg koristi i engleski jezik.