SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2977

Usporedba algoritama otkrivanja zajednica u društvenim mrežama

Daniel Marić

Zagreb, svibanj 2022.

Umjesto ove stranice umetnite izvornik Vašeg rada.
Da bi ste uklonili ovu stranicu obrišite naredbu `\izvornik`.

SADRŽAJ

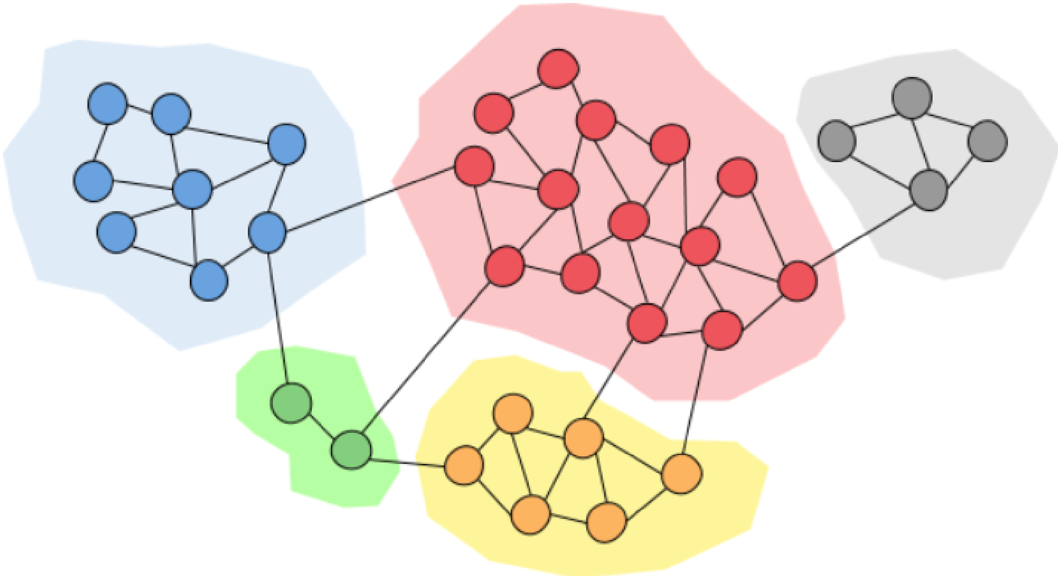
1. Uvod

Pojavom popularnih internetskih usluga za povezivanje korisnika stvorene su velike mreže društvenih zajednica. Generirane su velike količine podataka iz kojih je moguće izvući mnoštvo korisnih informacija. Takve zajednice sastoje se od puno manjih zajednica koje se po svojim karakteristikama razlikuju od ostalih. Zajednice je potrebno pronaći kako bi im se pristupilo na najbolji mogući način. Rješavanje ovog problema važno je i u drugim granama znanosti, kao na primjer u sociologiji, biologiji ili računarskoj znanosti, gdje su problemi predstavljeni na takav način, pomoću strukture grafa.

Upravo su grafovi najpogodnija struktura podataka za pristup ovome problemu, gdje relevantne značajke, u primjeru društvenih mreža, ljude možemo prikazati pomoću čvorova, dok će bridovi predstavljati veze između tih značajki. Više bridova među određenim značajkama značit će da tu mogu postojati obilježja zajednice, npr. u biologiji bi to mogla biti tkiva koja u organima obavljaju sličnu ulogu.

Rješavanje problema koji su predstavljeni grafovima je vrlo složeno, vremenski i prostorno. Ovakvi grafovi nisu jednostavnog oblika, ali u njima postoje određene pravilnosti, koje se mogu iskoristiti. U tu svrhu razvijeno je mnogo algoritama za otkrivanje zajednica koji različitim pristupima pokušavaju pronaći rješenje ovog problema. Pojedini algoritmi su bolji od drugih na jednom tipu društvenih mreža ili lošiji na drugom te se zato koriste evaluacijske mjere kojima se procjenjuje koliko je dobro rješenje koje je algoritam pronašao. Što više algoritama se testira s različitim društvenim mrežama i evaluacijskim mjerama dobit će se bolji uvid u to kada je koji bolje koristiti. Najpoznatiji algoritam među njima je Newman-Girvanov algoritam koji će se nešto detaljnije opisati uz još nekoliko njih. Sličnim problemima bavili su se radovi Lancichinettija i Fortunata [6] iz 2016. te [9] iz 2009. godine.

Unatrag nekoliko posljednjih godina uvedeni su zakoni o zaštiti osobnih podataka te je sada znatno teže dobiti pristup korisnim informacijama. Zato se



Slika 1.1: Primjer grafa nepreklapajućih društvenih zajednica.

koriste posebni algoritmi za generiranje umjetnih skupova podataka koji će za zadane parametre generirati grafovi pomoću kojih se mogu provoditi istraživanja.

U nastavku rada bit će opisana struktura i svojstva društvenih mreža i zajednica, algoritmi koji pronalaze društvene zajednice, skupovi podataka koji su korišteni u sklopu rada, programsko rješenje koje pokreće i evaluira rješenja algoritama te će se prikazati i rezultati do kojih se došlo.

2. Društvene mreže i zajednice

Društvene mreže može se pronaći gdje god postoji sustav koji sadrži entitete koji su međusobno povezani. Primjera je mnogo, a neki od njih su: društvene web platforme, email mreže, web stranice koje sadrže poveznice prema drugima, uređaji koji su povezani preko internetske mreže i slično. Kako bi se skupina entiteta mogla nazvati društvenom zajednicom među njima mora postojati nekakav tip odnosa. Odnos može biti jednosmjeran ili dvosmjeran te mogu postojati težine kojima se odnosu daje veća ili manja značajnost. Društvene mreže imaju složenu organizacijsku strukturu te se može pretpostaviti svojstvo lokalnosti, koje kaže da ako jedan entitet ima veze prema neka druga dva entiteta onda je vjerojatnost da ta druga dva entiteta imaju vezu, veća od prosječne vjerojatnosti.

Društvene mreže imaju karakteristično svojstvo grupiranja u strukturu zajednice. Ako se čvorovi mreže mogu podijeliti u nepreklapajuće ili preklapajuće zajednice tako da broj veza između članova zajednice značajno premašuje broj veza između bilo koje dvije zajednice, znači da mreža ima strukturu društvenih zajednica. Mreže koje imaju takvu strukturu često se mogu prikazati i kao hijerarijske strukture. U ovom radu obradit će se mreže koje sadrže nepreklapajuće strukture sa vezama koje nemaju određene težine.

Proces pronalaska društvenih zajednica jedan je od glavnih zadataka u analizama društvenih mreža. Detekcija zajednica može biti vrlo korisna u raznim primjenama kao što je primjerice pronalaženje grupa kojima bi se mogle slati reklame za određene proizvode koji bi ih mogli zanimati umjesto da se svakom pojedincu šalju posebno. Još jedan primjer bio bi preporuka određenih sadržaja koji bi se mogli prikazivati grupama koje pokazuju zanimanja prema sličnim interesima. Primjera ima još mnogo, ali iz ova dva već je vidljivo da se korisne informacije mogu zaključivati iz društvenih mreža. Kako bi društvene mreže pohranili i analizirali u računalu potrebna je prikladna struktura podataka koja će u ovom slučaju biti graf.

2.1. Reprezentacija društvenih mreža

Graf je važna struktura podataka u području računarstva. Pomoću njega moguće je prikazati razne odnose i procese područja bioloških, društvenih i informacijskih sustava. Grafovima se modeliraju vrlo teški problemi kao primjerice problem kineskog poštara ili problem trgovačkog putnika koji je NP težak problem što znači da nema rješenje u polinomnom vremenu.

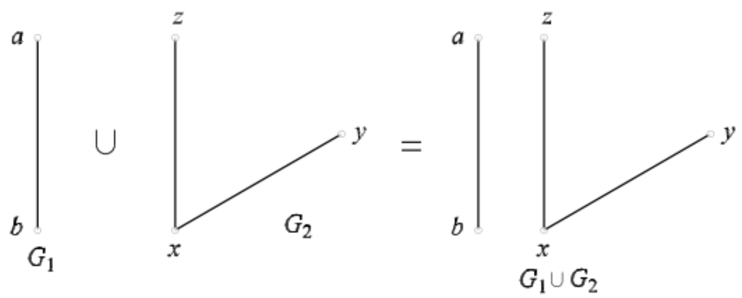
Prema definiciji jednostavan graf G sastoji se od nepraznog konačnog skupa *V(G)*, čije se elemente naziva vrhovi ili čvorovi grafa i konačnog skupa *E(G)* različitih dvočlanih podskupova skupa *V(G)* koji se naziva bridovima [11]. Graf može imati najviše $\frac{n(n-1)}{2}$ bridova. U radu će se razmatrati jednostavni grafovi koji nemaju petlje i više bridova između istih čvorova. Bridovi će biti bestežinski i neusmjereni.

Bitna definicija tiče se stupnja vrhova grafa. Stupanj vrha v grafa G je broj bridova koji su incidentni s v. Stupanj vrha označava se sa *deg(v)*. Vrh stupnja 0 zove se izolirani vrh, a vrh stupnja 1 krajnji vrh. [11]

Šetnja je graf sa skupom vrhova *V(G)* $= \{x_1, x_2, ..., x_l\}$ i bridova *E(G)* $= \{x_0x_1, x_1x_2, ..., x_{l-1}x_l\}$. vrhovi x_0 i x_l definiraju se kao krajevi dok je l duljina šetnje. Ako su svi bridovi šetnje različiti tada se ona naziva staza. Ako su uz to i svi vrhovi različiti onda se takva šetnju naziva putem. Ako put počinje i završava u istom vrhu tada graf sadrži ciklus. Uz pretpostavljena ograničenja najmanji ciklus koji graf u ovom radu može imati je trokut što je često obilježje društvenih mreža.

Definicija puta omogućava definiranje važnog koncepta koji će se pojavljivati u radu pojedinih algoritama. Ako u grafu za svaki par vrhova postoji barem jedan put koji ide od jednog do drugog vrha onda je graf povezan. Ako između vrhova postoji više putova onda je najkraći onaj koji ima najmanju duljinu. Promjer ili dijametar povezanog grafa je najveća udaljenost između bilo koja dva vrha u grafu. Ako ipak postoji barem jedan par vrhova između kojih ne postoji put onda je graf podijeljen u barem dva podgrafa. Svaki maksimalno povezani podgraf zove se komponenta povezanosti. Primjer se može vidjeti na slici 2.1.

Grafovi se mogu pohranjivati u obliku matrice susjedstva gdje su dva vrha, i i j susjedna ako im je element matrice Aij jednak 1, a inače 0. Zbog pretpostavke da ne postoje petlje na dijagonali matrice susjedstva svi su elementi nule. Za reprezentaciju neusmjerenog grafa matrica susjedstva je simetrična što znači da je dovoljno pohraniti samo jedan trokut matrice, iznad ili ispod dijagonale. Suma



Slika 2.1: Primjer nepovezanog grafa

elemenata i-tog retka ili stupca jednaka je stupnju vrha i

Jednostavniji oblik pohrane liste susjedstva koja se koristi tako da se pohranjuje skup susjednih bridova koji predstavljaju graf. Lista susjedstva je prostorno učinkovitija od matrice susjedstva kada su u pitanju rijetki grafovi kod kojih većina vrhova nije međusobno povezana. Prostorno zauzeće ovisi o broju vrhova i bridova u grafu, dok je kod matrice susjedstva uvijek proporcionalno kvadratu broja vrhova.

2.2. Obilježja društvenih zajednica

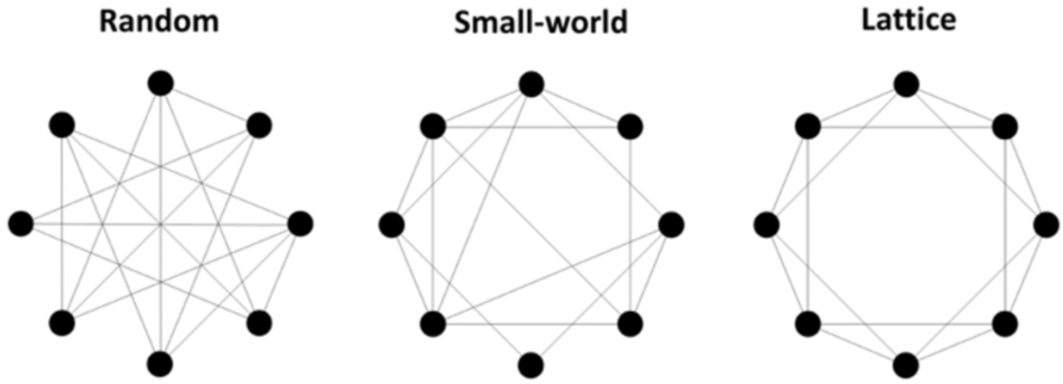
Društvene zajednice moguće je definirati na nekoliko načina sa različitih stajališta, ali ne postoji niti jedna univerzalno prihvaćena definicija. Definiranje vrlo često ovisi o problemu koji se promatra zajedno sa specifičnim detaljima i primjenama gdje se pojam zajednice koristi. Prema radu [5] zajednice je moguće promatrati iz lokalne i globalne perspektive.

Iz lokalne perspektive zajednica se može promatrati kao grupa entiteta koji su međusobno sličniji u odnosu na ostale entitete skupa podataka. Zajednica se formira tako što slični elementi imaju mnogo više interakcija sa članovima unutar zajednice u odnosu na one izvan. Zajednica se može smatrati kao autonomna skupina te ima smisla u određenim situacijama evaluirati svaku zasebno od ostatka društvene mreže. Stroga definicija društvene mreže kaže kako je društvena zajednica podgraf u kojem su svi članovi međusobno u interakciji [10]. Takva definicija odgovara terminu klike u teoriji grafova koji označava skup vrhova koji su svi međusobno susjedni. Najjednostavniji primjer klike je trokut i oni se pojavljuju u svim društvenim mrežama. Veće klike od trokuta se pojavljuju rjeđe te ovakva definicija tako postaje manje praktična u stvarnim primjerima. Još jedan problem klike je to što su tada svi vrhovi simetrični bez mogućnosti razlikovanja njihovih svojstava. U praktičnim primjerima očekuje se da među vrhovima postoji određena hijerarhijska struktura sa više i manje važnim čvorovima. Moguće je relaksirati pojam klike. Mogućnost je iskoristiti doseg i duljinu puta između čvorova. n-klika je takav podgraf da niti jedan par vrhova nije međusobno udaljen za više od n koraka i skup je maksimalan u smislu da niti jedan drugi čvor nije udaljen za više od n od svakog čvora iz podgrafa. Može se primijetiti da članovi podgrafa mogu biti povezani preko posrednika koji nije član grupe te onda n-klika ipak nije dovoljno dobra definicija. Definicija n-klana to popravlja. n-klan je n-klika u kojoj je dijametar podgrafa manji ili jednak n. Takva definicija ima problem što u njoj i dalje postoji zahtjev n-klike te se tako dolazi do definicije n-kluba. n-klub je podgraf gdje je dijametar manji ili jednak n. Tada je i svaki n-klan i n-klub i n-klika.

Iz globalne perspektive zajednica se može definirati promatrajući graf u cjelini. Takve definicije koriste se u slučajevima kada su zajednice dijelovi sustava bez kojih bi njegovo funkcioniranje bilo značajno izmijenjeno. Definicije se najčešće izvode indirektno, iz algoritama prema kojem je neko svojstvo iskorišteno kako bi se zajednice otkrile. Moguće je definirati null model koji će odgovarati prema određenim strukturnim karakteristikama, ali inače je slučajni graf. Model se tada koristi za usporedbu kako bi se odredilo ima li promatrani graf strukture zajednica. Poznati null model grafa predložili su Newman i Girvan koji se dobije tako da se u početnom grafu slučajno prespajaju bridovi pod uvjetom da stupanj svakog vrha ostane isti kao u početnom grafu. Iz njega je proizašla definicija modularnosti, odnosno funkcije kojom je moguće ocijeniti kvalitetu pronađenih zajednica u grafu. Modularnost je važna mjera jer ima nekoliko primjena u području otkrivanja zajednica. Koristi se kao mjera koja određuje koliko su kvalitetne pronađene grupe u mreži, ali i kao sastavni dio poznatog Girvan-Newmanovog algoritma [5].

2.3. Small-world mreže

Small-world mreže imaju obilježja dva tipa mreža. Prva mreža je slučajna mreža za koju je karakteristično što je prosječna udaljenost između dva vrha vrlo mala. Druga mreža je rešetkasta u obliku prstena gdje je svaki čvor susjedan sa $\frac{n}{2}$ čvorova sa svake strane. Small-world mreža posjeduje svojstva tih grafova te se pomoću njih može procijeniti u kojoj je mjeri mreža zaista small-world. Na temelju tih svojstava nastao je i Watts–Strogatz model koji služi za generiranje



Slika 2.2: Primjeri slučajnog, small-world i rešetkastog grafa.

slučajnih grafova društvenih mreža što se može iskoristiti u testiranjima raznih algoritama za detekciju zajednica. Primjeri mreža prikazani su na slici 2.2.

Small-world mreža je graf u kojem većina čvorova nisu susjedi, ali susjedi nekog čvora imaju veliku vjerojatnost da su i oni susjedi te se do svakog čvora može doći kroz nekoliko koraka što znači da bilo koja dva čvora imaju kratku međusobnu udaljenost. Specifično je što se ona za dva slučajno izabrana čvora te za fiksiran prosječan stupanj vrha povećava proporcionalno logaritmu broja čvorova u grafu dok koeficijent grupiranja nije malen. Small-world mreže sadrže klike i grupe koje su gotovo klike što proizlazi iz visokog koeficijenta grupiranja. Društvene mreže posjeduju svojstva small-world mreže.

Koeficijent grupiranja je mjera stupnja u kojem čvorovi u grafu teže grupiranju. Postoje dvije verzije mjere, lokalna i globalna. U lokalnoj verziji mjera se računa za pojedini čvor te govori u kolikoj je on mjeri grupiran sa svojim susjedima. Mjera se za čvor i računa kao suma broja veza koje postoje između susjeda promatranog čvora podijeljeno sa brojem svih mogućih veza,

$$C_i = \frac{2 \mid e_{jk} : v_j, v_k \in N_i, e_{jk} \in E \mid}{k_i(k_i - 1)}. \tag{2.1}$$

Ako iz formule maknemo koeficijent 2 tada se ona može koristiti za usmjerene grafove. Globalni koeficijent grupiranja daje informaciju o grupiranju u cijeloj društvenoj mreži. Temelji se na trojkama čvorova. Trojku čine promatrani čvor i druga dva čvora. Ako su povezani sa dva brida zovu se otvorena trojka, a ako su povezani sa tri zovu se zatvorena trojka što znači da jedan trokut čine tri trojke. Koeficijent se tada računa kao broj zatvorenih trojki podijeljen sa ukupnim brojem trojki,

$$C = \frac{broj\ zatvorenih\ trojki}{ukupan\ broj\ trojki}. \tag{2.2}$$

Formula je primjenjiva i na usmjerene i neusmjerene grafove.

Kratka prosječna duljina puta između čvorova znači da postoje čvorovi sa velikim brojem veza odnosno visokim stupnjem. Takvi čvorovi nazivaju se sabirnice te služe kao posrednici u mnogim putevima između ostalih čvorova. Primjer iz stvarnog svijeta može se pronaći u zračnim letovima između gradova. Na putovanju između dva grada vrlo često nije potrebno više od tri leta jer mnogo letova ide preko jednog velikog grada sa puno letova prema drugima.

Koliko mreža pripada small-world mreži može se izraziti pomoću small-koeficijenta, σ, koji se računa tako da se uspoređuju koeficijent grupiranja i karakteristična duljina puta u mreži sa slučajnim grafom koji ima jednak prosječan stupanj vrhova. Za karakterističnu duljinu puta najčešće se koristi prosječna minimalna udaljenost između vrhova. Koeficijent se računa prema formuli:

$$\sigma = \frac{\frac{C}{C_r}}{\frac{L}{L_r}}. \tag{2.3}$$

C i L su mjera grupiranja i prosječna duljina puta u promatranoj mreži dok su C_r i L_r su mjera grupiranja i prosječna duljina puta u slučajnom grafu. Ako je $\sigma > 1$ tada se može smatrati da je mreža small-world. No mjera pokazuje lošu otpornost na rast broja čvorova u mreži [14].

Druga mjera kojom se može izmjeriti koliko je mreža small-world uspoređuje promatranu mrežu s mrežom rešetkastog oblika (eng. lattice network) i slučajnom mrežom. Mjera kombinira karakterističnu duljinu puta i koeficijent grupiranja sa koeficijentom grupiranja rešetkaste mreže i karakterističnom duljinom puta ekvivalentnog slučajnog grafa prema sljedećoj formuli:

$$\omega = \frac{L_r}{L} - \frac{C}{C_l} \tag{2.4}$$

Ovakva definicija nije osjetljiva na mjeru C_r koja nije primjerena za mjerenje je li mreža small-world jer slučajni graf nema svojstva grupiranja. Vrijednosti koeficijenta ω ograničene su na interval između -1 i 1 bez obzira na veličinu mreže. Za vrijednost oko 0 može se smatrati da je mreža small-world što znači da je $L \approx L_r$ i $C \approx C_l$. Pozitivne vrijednosti ukazuju na to da graf ima više sličnosti sa slučajnim grafom, dok negativne na to da je graf pravilnijeg, rešetkastog oblika [14].

Posljednja mjera koja kvantificira small-world mjeru normalizira koeficijent grupiranja i duljinu puta mreže relativno u odnosu na karakteristike ekvivalentne rešetkaste i slučajne mreže. Small World Index (SWI) računa se na sljedeći način:

$$SWI = \frac{L - L_l}{L_r - L_l} \cdot \frac{C - C_r}{C_l - Cr} \tag{2.5}$$

Mjera ima interval rezultata između 0 i 1. Što je bliže 1 to je više vjerojatno da je mreža small-world. Vrlo je vjerojatno da ne postoji mreža koja bi imala SWI = 1, ali ideja mjere je izmjeriti small-world svojstvo na način koji bi teoretski činio mrežu idealnom small-world mrežom gdje vrijedi da je $C \approx C_l$ i $L \approx L_r$.

3. Algoritmi otkrivanja društvenih zajednica

Ključan dio u pronalasku društvenih zajednica u društvenim mrežama su algoritmi koji ih otkrivaju. Oni moraju biti pouzdani i učinkoviti, ali se i izvršavati u prihvatljivom vremenskom okviru. Algoritmi se testiraju na brojnim skupovima podataka uz prikladne evaluacijske mjere kako bi se zaključilo u kojim uvjetima koji algoritam daje najbolje rješenje.

Grafove koji predstavljaju društvene zajednice teško je prikazati u ravnini ako teže stvarnim veličinama koje se kreću u tisućama čvorova, a često i mnogo više, što znači da se ne može iz ljuske perspektive odrediti kako bi dobar raspored zajednica izgledao. To znači da su algoritmi koji pronalaze društvene zajednice nenadzirani algoritmi koji sami, bez primjera za učenje i prethodnog znanja o njima pokušavaju pronaći rješenje. U društvenim mrežama algoritmi koriste topološke karakteristike i specifičnosti koje posjeduju ovakvi tipovi mreža.

Dvije važne tehnike na kojima se temelji većina algoritama su particioniranje i grupiranje. Particioniranje grafova je proces u kojem se graf dijeli na unaprijed određeni broj manjih komponenti pomoću određenog svojstva. Svojstvo koje se može iskoristiti je minimalni rez. Ono se koristi tako da se graf podijeli na dva ili više razdvojenih podgrafova, a veličina reza koja se pokušava minimizirati je broj bridova koje je potrebno ukloniti da bi to ostvarili. Potrebno je odrediti i svojstvo koje bi odredilo veličinu komponenti kao primjerice minimalan ukupan stupanj vrhova kako bi se dobila rješenja koja imaju smisla. Zbog takvih zahtjeva ovakav pristup najčešće nije prihvatljiv jer broj zajednica nije moguće unaprijed odrediti. Grupiranje je proces u kojem se entitete koji imaju zajedničke karakteristike svrstava u iste grupe. Pronalaženje grupa može dati informacije o skrivenim značajkama, vezama i svojstvima članova te koliko su međusobno čvrsto povezani. U hijerarhijskom grupiranju stvara se hijerarhija među zajednicama. Proces se može odvijati na dva načina, aglomerativno ili divizivno. U aglomerativnom načinu koristi se pristup koji ide od dna prema vrhu te se određeni čvor dodaje drugim sličnim čvorovima uz korištenje određenog kriterija sličnosti. U divizivnom načinu veće grupe dijele se na manje uz korištenje određene mjere koja govori koliko je dobra trenutačna podjela prema kojoj će se odrediti konačan rezultat.

3.1. Girvan-Newmanov algoritam

Veliko zanimanje i rast aktivnosti znanstvene zajednice u području društvenih mreža potaknuo je rad [8] Girvana i Newmana iz 2002. godine u kojem su predstavili novi algoritam koji se po njima i naziva. Dotad poznati algoritmi pokušavali su pronalaženje zajednica riješiti tako da bi provodili hijerarhijsko grupiranje. U početnom koraku kreće se od nepovezanog grafa te se za svaki par vrhova računa težina koja predstavlja koliko su vrhovi bliski. Tada se bridovi se dodaju jedan po jedan počevši od onih vrhova čija je bliskost najveća. Postoji više načina kako izračunati bliskost i temelje se na broju puteva između čvorova, npr. broj vršno nezavisnih putova ili bridno nezavisnih putova i slično. Takve definicije ipak u nekim slučajevima nisu uspješne i daju krive rezultate. Događa se da se vrhovi koji su na rubovima zajednice, povezani jednim bridom prema ostatku mreže izdvajaju iz zajednice kojoj pripadaju i ostaju potpuno izolirani od svih zajednica. Girvan-Newmanov algoritam pokušava suprotno, pronaći bridove koji što manje doprinose povezanosti unutar zajednica.

Algoritam traži bridove koji povezuju zajednice te ih kroz iteracije uklanja i izolira zajednice. Za pronalazak bridova koristi se mjera različitosti, u ovom slučaju mjera bridne centralnosti. Njezina vrijednost računa se za svaki brid tako što se za sve parove vrhova odredi najkraći put te se svim bridovima koji se nalaze u tom putu dodaje vrijednost 1. Ako postoji N najkraćih putova između vrhova onda se u svim putevima svakom bridu vrijednost uvećava za $\frac{1}{N}$. Bridna centralnost svakog vrha na početku je postavljena na 0. Postupak se ponavlja dok god postoji bridova u grafu. Izračunavanje bridne centralnosti je skupa operacija jer je potrebno za svaki par vrhova u svakoj iteraciji pronaći najkraći put te odrediti bridne centralnosti. Mora se provoditi u svakom koraku jer se inače mogu dogoditi pogreške u koracima algoritma zato što se mreža prilagođava novom stanju nakon uklanjanja svakog brida. Takva situacija može se dogoditi ako su dvije zajednice povezane sa više bridova. Tada je, prema algoritmu, sigurno da će barem jedan od tih bridova imati visoku bridnu centralnost te se zato nakon njegovog uklanjanja vrijednost mjere mora ponovno odrediti, a onda će jedan od preostalih bridova imati najvišu vrijednost. Moguće je uštedjeti nešto resursa tako što se bridna centralnost izračunava samo za one vrhova na koje je uklanjanje prethodnog brida imalo utjecaja.

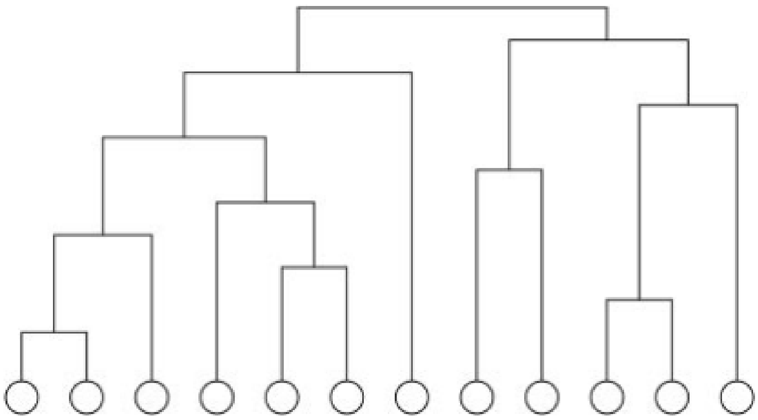
Algorithm 1 Girvan-Newmanov algoritam

1: izračunati mjeru različitosti za sve bridove u grafu
2: ukloniti brid sa najvećom vrijednosti mjere različitosti
3: za svaki brid izračunati mjeru različitosti nakon uklanjanja brida
4: ponavljati korake 2 i 3 dok ima bridova u grafu

Konačno rješenje algoritma određuje se tako što se u svakoj iteraciji računa modularnost za trenutnu podjelu grafa. Modularnost je mjera koja se koristi za procjenu jakosti veza unutar zajednice i jakosti veza među zajednicama. Njome je moguće izmjeriti koliko je određena podjela grafa kvalitetna. Ona podjela koja ima najvišu vrijednost na kraju algoritma uzima se kao rezultat. O modularnosti i ostalim mjerama za evaluiranje rezultata više će biti rečeno u poglavlju 6 koje se bavi vrednovanjem rezultata algoritama.

Postupak traženja zajednica tijekom rada algoritma može se predstaviti dendogramom. Dendogram je prema strukturi stablo gdje su listovi pojedini vrhovi mreže. Prema vrhu stabla vrhovi se spajaju u zajednice te konačno u cijelu strukturu grafa. Vrhovi povezani na nižim razinama imaju snažnije međusobne veze. Rez kroz stablo na bilo kojoj razini daje skup zajednica koji u tom trenutku postoji. Gdje stablo odrezati određuju se pomoću modularnosti. Primjer je prikazan na slici 3.1.

Za graf od m bridova i n vrhova složenost algoritma u najgorem slučaju je $\mathcal{O}(m^2n)$. Potrebno je u svakom koraku algoritma ponovno izračunati mjeru različitosti što ima velik utjecaj na povećanje složenosti. U boljim slučajevima, kada se mreža nakon nekoliko iteracija razdvoji u nekoliko komponenti izvođenje algoritma znatno se ubrzava. U nekoliko slučajeva isprobana je strategija da se mjera različitosti izračuna jednom, na početku izvođenja algoritma, ali u radu [8] pokazalo se da takav postupak ne daje ispravna rješenja.



Slika 3.1: Primjer hijerarhijskog stabla. Moguće je prema koracima vidjeti kada je koji brid uklonjen iz mreže te kako su grupe nastale.

3.2. Louvain algoritam

Algoritam osmišljen na sveučilištu u Louvainu stvoren je kako bi nadmašio dotad poznate algoritme u području otkrivanja zajednica. Objavljen je u radu [2], 2008. godine. Algoritam je testiran pronalaženjem zajednica u belgijskoj telefonskoj mreži od 2.6 milijuna korisnika te analiziranjem web grafa od 118 milijuna čvorova i više od milijardu bridova za koji je rješenje izračunao u 152 minute. Kapaciteti algoritma i testiranih mreža bili su ograničeni samo dostupnim kapacitetom računalnih resursa, a ne vremenom potrebnim za računanje. Algoritam se koristi heurističkom metodom koja se temelji na optimizacije modularnosti.

Rad algoritma podijeljen je u dvije faze koje se ponavljaju kroz iteracije. U prvom koraku, za graf od N čvorova, svaki čvor pripada zajednici u kojoj je samo on član što znači da u početku postoji onoliko zajednica koliko graf ima čvorova. Tada se za svaki čvor i promatraju njegovi susjedi j te se računa promjena modularnosti koja bi se dogodila ako se čvor i premjesti u zajednicu čvora j. Čvor i se premješta u zajednicu za koju je promjena maksimalna i pozitivna, odnosno modularnost raste. Ako nema rasta promatrani čvor ostaje u dodijeljenoj zajednici. Proces se ponavlja slijedno dok god postoje poboljšanja. Moguće je da se svi čvorovi razmatraju kroz nekoliko iteracija. Kada se u iteraciji ne dogodi niti jedno poboljšanje algoritam završava prvu fazu. U radu [2] pokazalo se kako redoslijed čvorova nema značajnijeg utjecaja na rezultat, ali se može uštedjeti na vremenu izračunavanja. Dio učinkovitosti algoritma proizlazi iz jednostavnog računanja promjene modularnosti tijekom premještanja čvora i u grupu C koja se dobiva prema sljedećem izrazu:

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]. \tag{3.1}$$

$\sum_{in}$ predstavlja sumu težina bridova, C, $\sum_{tot}$ je suma težina bridova koji su povezani sa čvorovima u zajednici C, k_i je suma težina bridova koji su incidentni sa čvorom i. $k_{i,in}$ je suma težina bridova koji povezuju čvor i sa zajednicom C i m je broj bridova u cijeloj mreži. U primjeni se promjena modularnosti računa tako što se čvor i premješta iz trenutne zajednice u susjedne te se promatra kako se vrijednosti ponašaju.

Algorithm 2 Louvain algoritam

Require: graf G

```
repeat
  svaki čvor grafa G dodijeliti u vlastitu zajednicu
  while postoje čvorovi koji se premještaju do
    for all čvor n grafa G do
      postaviti čvor u susjednu zajednicu uključujući i trenutnu tako da se
        rast modularnosti maksimizira
    end for
  end while
  if nova modularnost veća od prethodne then
    graf G postaje novi graf sa čvorovima koje čine prethodno dobivene za-
      jednice
  else
    kraj
  end if
until
```

Druga faza algoritma sastoji se od građenja nove mreže čiji čvorovi postaju zajednice koje su pronađene tijekom prve faze. Kako bi se dobio cjelovit graf potrebno je nove čvorove povezati bridovima. Brid se dodaje tako da spaja zajednice čiji su čvorovi bili susjedni te mu je težina suma težina tih bridova. Ako je brid bio unutar zajednice predstavlja se petljom koju je moguće izostaviti. U početnom koraku za graf koji nema određene težine bridova, težine se mogu postaviti na 1.

Kada druga faza završi ponavlja se prva faza korištenjem novonastale mreže. Ako se promotri rad algoritma te činjenica da je u početku svaki čvor jedna zajednica, onda se može zaključiti da se broj zajednica kroz iteracije smanjuje te je resursno najzahtjevnija prva iteracija. Faze se ponavljaju dokle god ima promjena u strukturi zajednica te modularnost ne postigne maksimum. Može se primijetiti da se kroz proces algoritma prirodno uključio pojam hijerarhije kada se manje zajednice spajaju u veće. Visina hijerarhije ovisi o broju iteracija algoritma te je uobičajeno manji broj.

Algoritam je jednostavan, intuitivan i jednostavan za implementaciju te radi nenadzirano. Složenosti je $\mathcal{O}(n \log n)$. Zbog pohlepne optimizacije, jednostavnog izračuna promjene modularnosti i naglog rasta broj zajednica brzo se izvršava što se dodatno ističe u mrežama su rijetke i koje imaju čvrste strukture zajednica. Postoji problem koji se događa zbog modularnosti koja ima poteškoćs u prepoznavanju manjih zajednica što se naziva rezolucijski limit. Njegov utjecaj ublažen je time što algoritam u početnom koraku kreće od situacije gdje je svaki čvor u svojoj zajednici te je vjerojatnost da će dvije različite zajednice biti spojene tako da se čvorovi premještaju jedan po jedan vrlo niska. Ako zajednice pokažu veliku bliskost mogu se spojiti kasnije nakon što se čvorovi u njima združe. Takvo ponašanje ističe se u slučaju klika koje konačno budu u jednoj zajednici, ali su razdvojene u početnim prolazima što znači da je moguće dobiti uvid u rješenja međukoraka algoritma te se krajnjem korisniku tako može pružiti uvid u promatranje zajednica na određenoj rezoluciji, odnosno hijerarhijskoj razini.

3.3. Surprise algoritam

Surprise algoritam uvodi novu mjeru kvalitete podjele zajednica u mreži. U radovima [2] i [7] prikazano je kako modularnost ima određenih nedostataka. Događa se problem s rezolucijom te se manje zajednice združuju s većima pri čemu može doći do situacije da povezanost čvorova unutar zajednice postaje slaba. Modularnost također posjeduje mnogo sličnih lokalnih maksimuma što znatno otežava pronalaženje globalnog. Moguća je i situacija u kojoj se za strukturno vrlo različite podjele zajednica dobivaju slične vrijednosti modularnosti. Mjera koja bi trebala riješiti navedene probleme naziva se surprise. Predstavljena je u radu [1] kao mjera za procjenu kvalitete pronađenih zajednica algoritama hijerarijskog grupiranja. U mnogim radovima mjera je korištena za ocjenjivanje rezultata raznih algoritama, ali u radu [7] je iskorištena kao dio algoritma.

U grafu sa K čvorova i n bridova podijeljenih u N_c zajednica takvih da je l bridova mreže povezuje vrhove iste zajednice surprise mjera definira se na sljedeći način:

$$S = -ln \sum_{j=l}^{min(M,m)} \frac{\binom{M}{j}\binom{F-m}{n-j}}{\binom{F}{n}}. \tag{3.2}$$

Mjera je hipergeometrijska distribucija gdje F predstavlja maksimalan broj mogućih bridova u mreži, $F = \frac{K(K-1)}{2}$ dok je M maksimalan mogući broj bridova unutar N_c zajednica gdje je broj čvorova unutar zajednice i, c_i. M se računa prema sljedećem izrazu:

$$M = \sum_{i=1}^{N_c} \frac{c_i(c_i - 1)}{2}. \tag{3.3}$$

Surprise mjeru može se zamisliti kao mjerenje koliko je malo vjerojatno, iznenađujuće, pronaći zajednicu sa l bridova unutar nje kao u promatranoj mreži. Postupak se može promatrati kao posuda sa F loptica gdje svaka loptica predstavlja jedan od mogućih bridova unutar mreže, gdje ih je M crveno koji predstavljaju moguće bridove unutar zajednice, a $F - M$ ih je plavih koji predstavljaju bridove između različitih zajednica. Iz posude se izvlači n loptica koje predstavljaju stvaran broj bridova u grafu. Suma s desne strane jednadžbe je vjerojatnost izvlačenja barem l crvenih loptica. Obično vrijedi da je $M << F$ te vjerojatnost izvlačenja crvene loptice može biti niska.

Surprise algoritam detekcije zajednica sličan je Louvain algoritmu. U početnom koraku mreža se podijeli u onoliko zajednica koliko ima čvorova. Tada algoritam pokušava premještanjem čvorova pronaći takvu raspodjelu koja će maksimizirati vrijednost surprise mjere. Razlika u odnosu na Louvain algoritam je što se premještanje čvorova može obavljati na tri načina. Dvije zajednice mogu biti spojene u jednu, jedan čvor se može premjestiti iz jedne u drugu zajednicu ili se čvor može izdvojiti u novu grupu. Druge dvije operacije mogu se pokrenuti rekurzivno za svaku zajednicu što algoritmu daje dodatnu snagu. Time je moguće otkriti manje podgrupe unutar zajednice te ih potencijalno izdvojiti u novu zajednicu ili pridružiti nekoj od postojećih. Operacije također pomažu algoritmu u izbjegavanju lokalnih maksimuma.

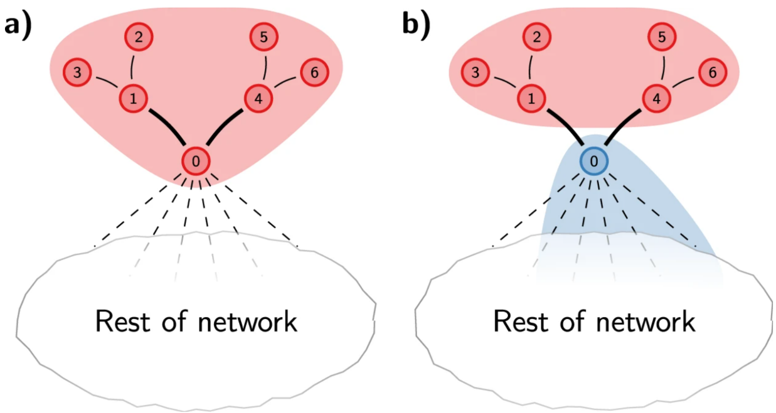
Algoritam je pohlepan što znači da prati svako poboljšanje vrijednosti mjere surprise, sve dok ona postoje. Istraživanja u radu [7] su pokazala da neke druge strategije u biranju smjera u kojem će surprise mjere rasti nisu pokazala bolje rezultate od pohlepne strategije dok su bile računalno zahtjevnije, no mogu se koristiti u završnim koracima algoritma kada razlike u rastu mjere postaju male.

Algorithm 3 Surprise algoritam

```
Require: mreža G
while postoje promjene u mreži do
    for svaka zajednica u mreži do
        for svaki čvor u zajednici do
            for svaki susjed čvora do
                if zajednica susjeda različita od trenutne zajednice then
                    pokušati spojiti zajednice ili izmijenti čvorove između njih
                end if
            end for
        end for
        while postoje promjene do
            izdvojiti čvor iz zajednice
        end while
        while postoje promjene do
            izdvojiti podzajednicu iz zajednice
        end while
        while postoje promjene do
            for svaku drugu zajednicu u mreži do
                pokušati premjestiti podzajednicu u drugu zajednicu
            end for
        end while
    end for
end while
```

Algoritam iterira kroz zajednice dok god se događaju promjene u strukturi zajednica. U svakoj zajednici prolazi kroz sve čvorove te ih pokušava pridružiti susjednima pojedinačno ili kao čitavu zajednicu. Potom pokušava iz svake zajednice izdvojiti čvor, izdvojiti podzajednicu ili je premjestiti u drugu zajednicu. Postojanje promjene znači da je određena operacija podigla vrijednost surprise mjere za novu raspodjelu.

Postoji velik broj kombinacija kako se čvorovi mogu podijeliti u zajednice te ih nije moguće sve provjeriti. Važno je promotriti kako se konfiguracija zajednica ponaša u situacijama kada su lokalni maksimumi slične vrijednosti te koliko su blizu globalnog. U radu [7] je pokazano da surprise algoritam ne pati od problema



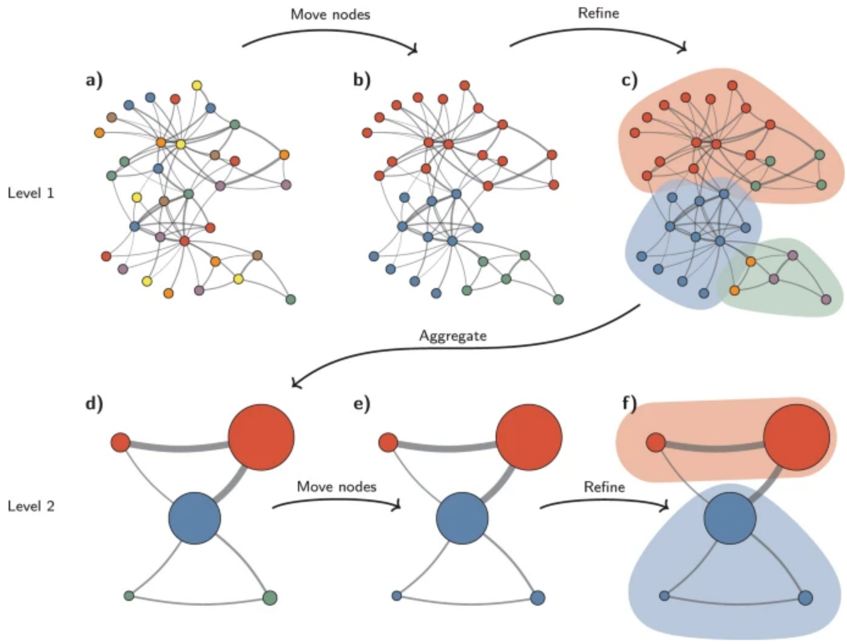
Slika 3.2: Primjer premještanja čvora iz zajednice u kojoj je predstavljao most između unutarnjih komponenti iz rada [16].

degeneracije ako je lokalni maksimum blizu globalnog što znači da i u slučajevima kada maksimum surprise mjere nije postignut konfiguracija mreže je reprezentativna što za modularnost nije vrijedilo. Ipak ne postoji garancija da će globalni maksimum biti pronađen, ali algoritam uz surprise mjeru trebao bi dati bolje rješenje u odnosu na onaj s modularnosti.

3.4. Leiden algoritam

Leiden algoritam osmišljen kako bi ispravio nedostatke Louvain algoritma. Lovain algoritam može kao konačan rezultat pronaći slabo povezane zajednice u situacijama gdje su one i iznad i ispod ranije spomenutog problema s rezolucijskom granicom. Algoritam može premjestiti čvor koji je predstavljao most između komponenti trenutne zajednice u drugu zajednicu dok ostali čvorovi ostaju u zajednici i ne premještaju se u neku od prikladnijih u tom trenutku. Događanje navedenog primjera izbjegava se u slučajevima kada je zajednica iznutra čvrsto povezana. Primjer opisane situacije nalazi se na slici 3.2. U a) dijelu čvorovi od 0 do 6 su u istoj zajednici te kada se čvor 0 premjesti zajednica prestaje biti povezana. Tada bi se dvije manje zajednice trebale moći izdvojiti u zasebne. Problem nastaje kada je zajednica lokalno optimalna te algoritam provodi drugu fazu u kojoj agregira zajednicu u jedan čvor što je najgori slučaj. Tada više nije moguće raditi premještanja unutarnjih čvorova te zajednica ostaje nepovezana osim ako se slučajno dogodi da se poveže sa nekom zajednicom koja će poslužiti kao novi most.

Leiden algoritam predstavljen je u radu [16] 2019. godine te pokušava is-



Slika 3.3: Ilustracija dvije iteracije Leiden algoritma iz rada [16]. U a) dijelu čvorovi se podijele u zasebne zajednice te se na slici b) spajaju u zajednice. Na c) slici provodi se proces pročišćavanja zajednica i proces kreće u novu iteraciju.

praviti nedostatke Louvaina korištenjem raznih, već poznatih strategija prilikom premještanja čvorova. Algoritam pametnog premještanja čvorova predstavljen je u radu [17]. Leiden koristi ideju o ubrzavanju premještanja lokalnih iz rada [12] te ideju o premještanje čvora u zajednicu slučajno izabranog susjednog čvora iz rada [15]. Algoritam, kao i Louvain, radi premještanja na temelju optimizacije modularnosti.

Leiden algoritam sastoji se od tri faze: lokalno premještanje čvorova, pročišćavanje zajednica i agregacija mreže temeljena na pročišćenim zajednicama. U Louvain algoritmu agregacija zajednica u nove čvorove stvara se na temelju trenutnih zajednica. Leiden algoritam pokušava pročistiti pronađene zajednice prije koraka agregacije. Tim korakom zajednica se može podijeliti u podzajednice koje će bolje predstavljati strukturu mreže. Pročišćene zajednice tada se agregiraju u čvorove za iduću iteraciju algoritma. Podjela zajednica ostaje ista kakva je bila prije faze pročišćavanja, kao u Louvain algoritmu. Implementacijom faze pročišćavanja rastu šanse da će algoritam pronaći zajednice visoke kvalitete i povezanosti.

Pročišćavanje zajednica provodi se na sličan način kao i prva faza algoritma, ali na razini zajednice. Svakom čvoru se dodijeli vlastita zajednica te se čvorovi lokalno spajaju sa drugima unutar zajednice ako su njihove veze dovoljno jake. Tijekom procesa pročišćavanja čvorovi se ne premještaju nužno pohlepno nego mogu biti dodani bilo kojoj zajednici za koju vrijednost funkcije kvalitete raste. Zajednica u koju će se čvor dodati odabire se slučajno, tako da je veća što je veći rast funkcije kvalitete. Razina slučajnosti ovisi o parametru $\theta > 0$. Slučajnost doprinosi istraživanju prostora zajednica i omogućava izlaženje iz potencijalnih lokalnih maksimuma. Isključivanje premještanja koja bi negativno utjecala na funkciju kvalitete nisu dozvoljena jer bi previše usporavala algoritam na velikim mrežama.

Još jedna razlika u odnosu na Louvain algoritam događa se u procesu lokalnog premještanja čvorova. Leiden algoritam koristi implementaciju brzog premještanja. Algoritam posjećuje čvorove dok god postoje promjene u funkciji kvalitete. U ovakvom pristupu posjećuju se samo oni čvorovi kojima se dogodila promjena u susjedstvu za razliku od Louvaina koji u svakoj iteraciji obilazi sve čvorove. Proces započinje tako da se u red dodaju svi čvorovi mreže slučajnim poretkom. Tada se uzima prvi čvor te se provjerava raste li funkcija kvalitete premještanjem u neku zajednicu. Ako se čvor premjesti, na kraj reda dodaju se svi susjedi čvora koji ne pripadaju novoj zajednici, a nisu već u redu. Proces se nastavlja dokle god red nije prazan. Korištenjem brzog premještanja prva iteracija je ista kao u Louvain algoritmu i obilaze se svi čvorovi, ali se u kasnijim iteracijama posjećuju samo oni čvorovi na koje promjene zaista utječu čime Leiden algoritam postaje značajno efikasniji.

Algorithm 4 Leiden algoritam

Require: mreža G

```
repeat
    lokalno premještanje čvorova
    if svaka zajednica se sastoji od samo jednog čvora then
        break
    else
        pročistiti zajednice
        agregirati mrežu na temelji pročišćenih zajednica
        zadržati određeni raspored zajednica
    end if
until
```

3.5. Walktrap algoritam

Walktrap algoritam predstavlja novu mjeru kojom računa sličnosti između čvorova i zajednica. Mjera se temelji na slučajnim šetnjama kroz mrežu koje imaju dobra svojstva u smislu računalne složenosti i otkrivanja strukture zajednica. Algoritam koristi definiciju zajednice koja kaže kako su grafovi društvenih mreža globalno rijetki, ali lokalno gusti. Gusti dijelovi nazivaju se zajednice koje su čvrsto povezane s nekoliko veza prema ostatku mreže. Ideja algoritma temelji se tako što bi slučajne šetnje kroz graf trebale ostati zarobljene unutar gustih dijelova, odnosno zajednica. Pomoću njih definiraju se mjere kojima se mogu mjeriti strukturalne sličnosti između čvorova i između zajednica. Dobiveni rezultati mogu se prikazati u hijerarhijskom obliku dendograma. Algoritam je predstavljen u radu [13].

Proces slučajne šetnje kroz graf G odvija se tako što slučajni šetač prelazi iz jednog čvora u drugi koji je izabran slučajno i uniformno u odnosu na ostale susjedne čvorove. Niz čvorova koji se posjećuju može se predstaviti Markovljevim lancem gdje su stanja vrhovi grafova. Vjerojatnost prijelaza iz čvora i u j u svakom koraku je $P_{i,j} = \frac{A_{i,j}}{d(i)}$, gdje je $A_{i,j}$ element u matrici susjedstva koji određuje jesu li čvorovi susjedni i iznosi 0 ako nisu ili 1 ako jesu, a $d(i)$ je stupanj čvora i. Isto se može zapisati kao matrica prijelaza slučajne šetnje P te se računa pomoću izraza $P = D^{-1}A$ gdje je D dijagonalna matrica stupnjeva vrhova, a A matrica susjedstva. Proces slučajne šetnje odvija se potenciranjem matrice P. Vjerojatnost prijelaza iz čvora i u j kroz t koraka je $(P^t)_{i,j}$.

Vjerojatnost zadovoljava dva svojstva karakteristična za proces slučajne šetnje. Prvo se odnosi vjerojatnost da se šetač nađe u čvoru j ovisi samo o njegovom stupnju kada duljina šetnje t teži prema beskonačnosti:

$$\forall i, \lim_{t \to +\infty} P^t_{i,j} = \frac{d(j)}{\sum_k d(k)}, \tag{3.4}$$

gdje $\sum_k d(k)$ predstavlja sumu vrhova svih čvorova u grafu. Drugo svojstvo kaže kako je vjerojatnost prelaska iz čvora i u j i iz j u i tijekom slučajne šetnje t ima omjer koji ovisi samo u stupnjevima čvorova:

$$\forall i, \forall j, d(i)P^t_{i,j} = d(j)P^t_{i,j}. \tag{3.5}$$

Dokazi su navedeni u radu [13].

Mjera udaljenosti r osmišljena je kako bi mjerila sličnost između čvorova u grafu. Udaljenost mora biti velika ako čvorovi pripadaju različitim zajednicama i mala ako su unutar iste. Računa se pomoću informacija dobivenih iz slučajnih šetnji. Za slučajnu šetnju duljine t kojom se dobiva informacija o vjerojatnosti prijelaza iz čvora i u j bitno je da duljina t dovoljno duga kako bi se prikupilo dovoljno informacija o topologiji mreže, ali duljina ne smije biti prevelika kako bi se izbjeglo prethodno navedeno svojstvo u izrazu 3.4 te vjerojatnost ne bi ovisila samo o stupnju čvora u koji se ide. Svaka vjerojatnost $P^t_{i,j}$ nosi određene informacije o čvorovima i i j, a svojstvo 3.5 govori kako vjerojatnosti $P^t_{i,j}$ i $P^t_{j,i}$ nose istu količinu informacije što se može iskoristiti u definiranju potrebne mjere. Kako bi se mjera ispravno koristila potrebno je istaknuti nekoliko činjenica. Ako je iznos vjerojatnost $P^t_{i,j}$ visok ne mora nužno značiti da su čvorovi i i j unutar iste zajednice. Na vjerojatnost $P^t_{i,j}$ utječe stupanj čvora $d(j)$ zato što se vjerojatnost da će slučajna šetnja u njemu završiti povećava što je stupanj veći. Konačno ako su dva čvora u istoj zajednici trebali bi ostatak mreže promatrati na isti način što znači da ako su i i j u istoj zajednici vrijedi $\forall k, P^t_{i,k} \approx P^t_{j,k}$, gdje je k bilo koji čvor u mreži. Mjera udaljenosti između dva čvora definira se sljedećim izrazom:

$$r_{i,j} = \sqrt{\sum_{k=1}^{n} \frac{(P^t_{ik} - P^t_{jk})^2}{d(k)}} \tag{3.6}$$

Na sličan način moguće je generalizirati udaljenost između čvorova na udaljenost između zajednica. Slučajna šetnja počinje iz jednog od slučajno, uniformno odabranih čvorova zajednice. Vjerojatnost $P^t_{C,j}$ šetnje iz zajednice C do čvora j se računa:

$$P^t_{C,j} = \frac{1}{|C|} \sum_{i \in C} P^t_{i,j}, \tag{3.7}$$

gdje je $|C|$ broj čvorova unutar promatrane zajednice. Mjera udaljenosti između dvije zajednice tada se definira na sljedeći način pri čemu vrijede ista načela kao i u definiciji udaljenosti između čvorova:

$$r_{C_1 C_2} = \sqrt{\sum_{k=1}^{n} \frac{(P^t_{ik} - P^t_{jk})^2}{d(k)}} \tag{3.8}$$

Algoritam započinje podjelom svih vrhova u zasebne zajednice. Kroz n - 1 koraka prema kriteriju udaljenosti spaja po dvije zajednice i dodaje ih u mrežu. Nakon dodavanja nove zajednice u mrežu računaju se nove udaljenosti među zajednicama, ali samo onima koji su u susjedstvu nove. Konačna struktura može se prikazati dendogramom u kojem se može vidjeti u kojem koraku su određene zajednice spojene.

Algorithm 5 Walktrap algoritam

Require: mreža G

1: podijeliti graf u n zajednica
2: **for** k od 1 do $n-1$ **do**
3: izabrati zajednice C_1 i C_2 prema kriteriju udaljenosti između zajednica
4: spojiti odabrane zajednice u novu $C_3 = C_1 \cup C_1$, dodati je grafu i izbaciti stare
5: izračunati promjenu u udaljenosti između zajednica
6: **end for**

Središnji dio algoritma predstavlja izračun kvalitete strukture zajednica. Kako bi se smanjila kompleksnost algoritma moguće je spajati samo susjedne zajednice. Zajednice koje će se spojiti biraju se prema Wardovoj metodi minimalne varijance. U svakom od k koraka spajaju se dvije zajednice koje minimiziraju srednju vrijednost kvadratne udaljenosti između svakog čvora i njegove zajednice.

$$\sigma_k = \frac{1}{n} \sum_{C \in P_k} \sum_{i \in C} r_{iC}^2 \tag{3.9}$$

Pristup je pohlepan i pokušava riješiti problem tako da uzima najbolje rješenju u koraku k. Problem je NP-težak i poznat je iz drugih algoritama grupiranja, npr. K-Medijana u radu [4], koji su ga pokušali iskoristiti, ali aproksimacije su bile eksponencijalne složenosti. Aproksimacijska promjena varijance računa se tako da se za svaki par susjednih zajednica izračuna promjena $\Delta\sigma(C_1C_2)$ koja bi se dogodila kada bi se zajednice C_1 i C_2 spojile u novu zajednicu C_3. Rezultat ovisi samo o čvorovima zajednica C_1 i C_2. Računa se prema sljedećem izrazu:

$$\Delta\sigma(C_1C_2) = \frac{1}{n}\left(\sum_{i \in C_3} r_{iC_3}^2 - \sum_{i \in C_1} r_{iC_1}^2 - \sum_{i \in C_2} r_{iC_2}^2 \right) \tag{3.10}$$

Konačno, spajaju se dvije zajednice za koje je $\Delta\sigma$ najmanje vrijednosti.

Vremenska složenost algoritma u najgorem scenariju je $\mathcal{O}(mn^2)$, a prostorna $\mathcal{O}(n^2)$, gdje je n broj vrhova, a m broj bridova grafa. No prema radu [13] u kojem je algoritam predstavljen, za rijetke grafove kakvi su grafovi društvenih mreža vremenska složenost je $\mathcal{O}(n^2 \log n)$, dok prostorna ostaje ista.

4. Skupovi podataka

U današnjem svijetu informacijski sustavi stvaraju goleme količine podataka. Podatke se može iskoristiti kako bi se poboljšali procesi koji se prate, pronašla područja u kojima postoji prostor za napredak, napraviti iskorak u poslovanju gdje se sustav primjenuje ili poboljšati korisničko iskustvo. Podaci su skupovi vrijednosti koji opisuju objekte u nekom procesu. Prate se kako bi se apstraktan proces iz ideje pretvorio u konkretne činjenice. Mogu se mjeriti, obrađivati i analizirati te vizualizirati kroz grafove, tablice i slike iz čega se dalje mogu izvoditi određeni zaključci.

Algoritme koji se razvijaju da bi u podacima pronalazili korisne informacije potrebno je evaluirati te ocijeniti kako se ponašaju na kojem skupu podataka i pronaći situacije u kojima pokazuju najbolje performanse. Podaci prikupljeni iz stvarnih sustava najbolje opisuju praćene procese te je se najbolji zaključci mogu donijeti koristeći upravo njih.

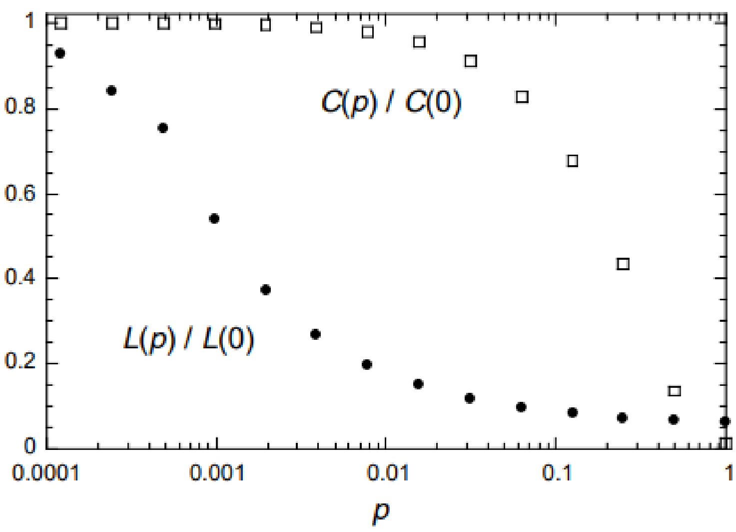
Nakon problema i greški tijekom prikupljanja i praćenja osobnih podataka te loše sigurnosti i slučajeva krađe podataka, 2018. godine uvedena je uredba o općoj zaštiti podataka, poznata pod kraticom GDPR. Uredbom se kontrolira pohrana, prijenos i obrada osobnih podataka u Europskoj Uniji te su navedeni procesi znatno postroženi. Nakon Europske Unije slične odredbe primijenile su i neke američke savezne države te neke Azijske države čime odredba počinje vrijediti u gotovo svim razvijenijim dijelovima svijeta. Time je područje analiza društvenih mreža značajno pogođeno jer je postalo mnogo teže dobiti podatke o stvarnim korisnicima. Od tada značajnu ulogu počinju imati algoritmi za generiranje mreža koje imaju karakteristike društvenih zajednica. Algoritmi u početnom koraku dobivaju određene podatke o veličini i svojstvima željene mreže te generiraju takav primjer. U nastavku poglavlja opisat će se Watts-Strogatz model koji generira umjetne skupove podataka te stvarni skupovi podataka pomoću kojih će se algoritmi navedeni u radu testirati.

4.1. Watts - Strogatz model

Watts-Strogatz model predstavljen je u radu [18] 1998. godine. Autori su pokušali stvoriti algoritam koji će stvoriti umjetnu mrežu, koja će imati svojstva small-world mreža koje se mogu pronaći u primjenama u biološkim, tehnološkim i društvenim sustavima. Mreže takvih sustava nalaze se između dva ekstrema, slučajne mreže i pravilne rešetkaste mreže. Kako bi se generirala mreža koja je između ta dva slučaja može se iskoristiti proces slučajnog prespajanja bridova mreže.

Proces započinje od pravilnog grafa prstenastog oblika sa n vrhova i k bridova po svakom vrhu te se svaki vrh prespaja s vjerojatnosti p. Vjerojatnost pruža mogućnost podešavanja oblika željenog grafa između regularnog za $p = 0$ i slučajnog za $p = 1$. Strukturna svojstva generiranog grafa mjere se pomoću karakteristične duljine puta, L i koeficijenta grupiranja C. L najčešće mjeri prosječnu duljinu puta u grafu i smatra se globalnim svojstvom grafa dok C mjeri koliko su susjedne veze jake te se smatra lokalnim svojstvom. Za graf kada $p \to 1$ pokazuje se da normalizirane vrijednosti L i C teže prema 1 , dok kada $p \to 0$ L i C teže prema 0 što bi moglo značiti da je velika vrijednost C povezana s velikim L, a mala vrijednost C sa malom vrijednosti L. No graf na slici 4.1 pokazuje kako se vrijednosti L i C mijenjaju u ovisnosti o vjerojatnosti p u odnosu na broj čvorova grafa. Vidljivo je kako postoji interval u kojem je L malen gotovo kao L_{random} dok je $C >> C_{random}$. Takva svojstva small-world mreže moguća su zbog neposrednog pada karakteristične duljine puta L uvođenjem tek nekoliko slučajnih bridova. Efekt je visoko nelinearan te se za male vrijednosti p brzo smanjuje udaljenost između vrhova, ali i zajednica. Za koeficijent grupiranja prespajanje ima tek linearan utjecaj te se koeficijent gotovo ne mijenja.

U radu[18] ideja je provjerena na mnogo različitih početnih oblika mreža iz čega su nastali i podaci za graf 4.1. Jedini uvjet je da bridovi koji se prespajaju moraju uglavnom povezivati vrhove za koje bi udaljenost bila mnogo veća nego u slučajnom grafu. Karakteristična duljina puta, $L(p)$ je definirana kao prosječna najkraća udaljenost između svih parova vrhova grafa. $C(p)$, koeficijent grupiranja, definiran je tako da ako čvor v ima k_v susjeda, onda među susjedima može postojati najviše $\frac{k_v(k_v-1)}{2}$ bridova. Tada se C_v definira kao omjer ostvarenih bridova među susjedima i ukupnog mogućeg broja bridova. Konačno C se računa kao srednja vrijednost C_v za svih v vrhova grafa. U društvenim mrežama navedene mjere imaju smisleno značenje. L je prosječan broj prijateljstava u lancu koji



Slika 4.1: Karakteristična duljina puta L i koeficijent grupiranja u grafovima s prespajanjem bridova u odnosu na normalizirane vrijednosti $L(0)$ i $C(0)$ za početni regularni graf.

povezuje bilo koja dva člana zajednice. C predstavlja koliko su prijatelji nekog člana zajednice povezani. Podaci prikazani u grafu 4.1 su uprosječeni rezultati 20 realizacija procesa slučajnog prespajanja vrhova grafa. Grafovi su imali po 1000 vrhova i prosječan stupanj vrha 10 po vrhu. Na horizontalnoj skali upotrijebljena je logaritamska skala kako bi se uspješno prikazao velik pad L, dok C ostaje konstantan kroz gotovo cijeli isječak koji graf prikazuje te ukazuje na činjenicu da je prelazak iz regularnog grafa prema small-world grafu gotovo neprimjetan na lokalnoj razini.

Algoritam generiranja modela grafa društvene zajednice sa small-world svojstvima odvija se kroz dva koraka. Algoritam prima tri parametra: N predstavlja broj vrhova grafa, k je srednja vrijednost stupnja pojedinog čvora i najčešće se odabire kao neki cijeli paran broj te p što je parametar slučajnosti prespajanja promatranog brida grafa. Na konkretne parametre postoje određeni uvjeti. Vjerojatnost p mora biti u granicama $0 \leq p \leq 1$. Parametri N i K moraju biti u sljedećem odnosu: $N >> K >> \log N >> 1$. Uvjetom $K >> \log N$ garantira se da će generirani graf biti povezan [3]. Algoritam konstruira neusmjereni graf sa N čvorova i $\frac{NK}{2}$ bridova na sljedeći način:

1. Konstruira se neusmjeren, regularan graf sa N čvorova povezan sa K susjeda, $\frac{K}{2}$ sa svake strane.

2. Za svaki vrh promatra se svaki brid kojim je spojen s $\frac{K}{2}$ desnih susjeda

i prespaja se s vjerojatnošću p. Prespajanje se obavlja tako da se brid spoji sa nekim od preostalih v vrhova, različitih od trenutno promatranog i onoga s kojim ga je brid u tom trenutku povezivao, te između promatranog i odabranog vrha već ne postoji brid.

4.2. Stvarni podaci

5. Programsko ostvarenje

...

6. Vrednovanje i rezultati

...

7. Zaključak

...

Literatura

[1] Rodrigo Aldecoa i Ignacio Marín. Jerarca: Efficient analysis of complex networks using hierarchical clustering. *PloS one*, 5(7):e11585, 2010.

[2] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, i Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[3] Béla Bollobás. *Random graphs*. Broj 73 u Cambridge studies in advanced mathematics. Cambridge University Press, 2 izdanju, 2001. ISBN 0-521-80920-7.

[4] W Fernandez De La Vega, Marek Karpinski, Claire Kenyon, i Yuval Rabani. Approximation schemes for clustering problems. U *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, stranice 50–58, 2003.

[5] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3-5): 75–174, 2010.

[6] Santo Fortunato i Darko Hric. Community detection in networks: A user guide. *Physics reports*, 659:1–44, 2016.

[7] Daniel Gamermann i José Antônio Pellizzaro. An algorithm for network community structure determination by surprise. *Physica A: Statistical Mechanics and its Applications*, 595:127063, 2022.

[8] Michelle Girvan i Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12): 7821–7826, 2002.

[9] Andrea Lancichinetti i Santo Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[10] R Duncan Luce i Albert D Perry. A method of matrix analysis of group structure. *Psychometrika*, 14(2):95–116, 1949.

[11] Anamari Nakić i Mario Osvin Pavčević. *Uvodna poglavlja u teoriju grafova.* UNIZG-FER, 2019.

[12] Naoto Ozaki, Hiroshi Tezuka, i Mary Inaba. A simple acceleration method for the louvain algorithm. *International Journal of Computer and Electrical Engineering*, 8(3):207, 2016.

[13] Pascal Pons i Matthieu Latapy. Computing communities in large networks using random walks. U *International symposium on computer and information sciences*, stranice 284–293. Springer, 2005.

[14] Qawi K Telesford, Karen E Joyce, Satoru Hayasaka, Jonathan H Burdette, i Paul J Laurienti. The ubiquity of small-world networks. *Brain connectivity*, 1(5):367–375, 2011.

[15] Vincent A Traag. Faster unfolding of communities: Speeding up the louvain algorithm. *Physical Review E*, 92(3):032801, 2015.

[16] Vincent A Traag, Ludo Waltman, i Nees Jan Van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports*, 9(1): 1–12, 2019.

[17] Ludo Waltman i Nees Jan Van Eck. A smart local moving algorithm for large-scale modularity-based community detection. *The European physical journal B*, 86(11):1–14, 2013.

[18] Duncan J Watts i Steven H Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

Usporedba algoritama otkrivanja zajednica u društvenim mrežama

Sažetak

Sažetak na hrvatskom jeziku.

Ključne riječi: Ključne riječi, odvojene zarezima.

Title on english

Abstract

Abstract.

Keywords: Keywords.