

UNIVERZITET U BEOGRADU
ELEKTROTEHNIČKI FAKULTET



ANALIZA SOCIJALNIH MREŽA

Projektni zadatak

Verzija 1.0

Predmetni nastavnici:

dr Marko Mišić, vanredni profesor

dr Jelica Protić, redovni profesor

Školska godina:

2025/2026.

Predmetni saradnik:

Predrag Obradović, asistent

Beograd, januar 2026.

SADRŽAJ

SADRŽAJ	2
1. UVOD	3
2. CILJ	3
3. POSTAVLJENI PROBLEM	3
3.1. ANALIZA MREŽA SLIČNOSTI U PROGRAMSKOM KODU.....	3
3.2. SKUP PODATAKA ZA ANALIZU	5
3.3. MODELOVANJE MREŽA.....	6
3.4. ISTRAŽIVAČKA PITANJA I CILJEVI	7
3.4.1. <i>Osnovna karakterizacija modelovanih mreža (10/20 poena)</i>	8
3.4.2. <i>Analiza mera centralnosti (4/20 poena)</i>	9
3.4.3. <i>Detekcija komuna Luvenskom metodom (2/20 poena)</i>	9
3.4.4. <i>Detekcija komuna spektralnim klasterisanjem (2/20 poena)</i>	9
3.4.5. <i>Poređenje modelovanih mreža (2/20 poena)</i>	9
3.5. PREPORUČENE METODE I ALATI.....	10
4. REZULTATI	10
5. PREDAJA, ODBRANA I VREDNOVANJE	10
LITERATURA	11

1. UVOD

U okviru ovog dokumenta su data uputstva za izradu projektnog zadatka na predmetu Analiza socijalnih mreža (13M111ASM) u školskoj 2025/2026. godini. Studenti treba da pažljivo pročitaju ovo uputstvo pre izrade projektnog zadatka. Studenti projektni zadatak rade **samostalno** ili **u paru**.

2. CILJ

Cilj projektnog zadatka na predmetu Analiza socijalnih mreža je praktična primena stečenog teorijskog znanja iz predmeta na primeru jednog konkretnog istraživačkog problema. Kroz zadati istraživački problem, studenti treba da izvrše prikupljanje, obradu i preliminarnu analizu primarnog (sirovog) skupa podataka, izdvoje neophodne podatke i modeliraju problem mrežom odgovarajućeg tipa. Modeliranu mrežu treba da analiziraju alatima za obradu socijalnih mreža po izboru i izvrše vizuelizaciju mreže. Dobijene rezultate analize treba na odgovarajući način interpretirati u skladu sa postavljenim istraživačkim pitanjima i predstaviti u obliku izveštaja.

3. POSTAVLJENI PROBLEM

U okviru ove sekcije je dat predlog projektnog zadatka za tekuću školsku godinu. Studenti mogu predložiti predmetnom nastavniku drugu temu. U tom slučaju, poželjno je priložiti i deo skupa podataka koji bi se analizirao, kako bi student na adekvatan način u saradnji sa nastavnikom postavio ciljeve istraživanja i istraživačka pitanja.

3.1. Analiza mreža sličnosti u programskom kodu

Pojava plagijarizma, odnosno predstavljanja tuđeg autorskog dela kao sopstvenog, predstavlja veliki problem današnjeg društva. Problem plagijarizma se često javlja i kod pisanja softvera, kako u softverskoj industriji, tako i u obrazovnom procesu na univerzitetima. Obrazovanje u oblasti računarstva uključuje značajan praktičan rad kroz programske zadatke različitog tipa koji su česta meta plagijarizma [1].

Plagijarizam u programskom kodu se može posmatrati kao svaka namerna upotreba programskog koda koji je napisao neko drugi, a koja nije adekvatno citirana od strane onoga ko

predaje programski kod kao svoje delo. U smislu detekcije ove pojave, bitno je eksplicitno razlikovati termine sličnost i plagijat. Detektovana sličnost između dva programska koda ne mora nužno biti rezultat plagijarizma. Pored kopiranja tuđeg programskog koda, postoji niz faktora koji mogu da utiču na pojavu sličnosti: upotreba uobičajenih algoritama i šablonskog koda, konvencije imenovanja identifikatora, delovi koda nastali korišćenjem alata za automatsko generisanje koda ili velikih jezičkih modela i sl. [2]

Pojava plagijarizma je dovela do razvoja alata za automatizovanu detekciju sličnosti, sa ciljem da se uporedi veći broj studentskih rešenja (programa) i olakša uočavanje međusobnih preklapanja od strane nastavnog osoblja i verifikaciju pojave plagijarizma. U tom smislu, alati za detekciju sličnosti barataju sa većim brojem kraćih izvornih kodova koji predstavljaju studentska rešenja i kvantifikuju meru sličnosti između pojedinačnih parova rešenja podnetih na proveru.

Alati za detekciju sličnosti najčešće poseduju tekstualni prikaz rezultata u obliku liste parova studentskih radova rangiranih po procentu sličnosti ili sličnoj metrici. Međutim, u realnim situacijama se može detektovati međusobna sličnost između više različitih radova, jer je plagijarizam društveni fenomen koji često obuhvata veći broj učesnika. Pritom, sami alati često ne daju više mogućnosti za dublju analizu i klasterizaciju rezultata. Stoga je prirodno rezultate poređenja prikazati i analizirati u obliku kolaborativne mreže (grafa). Za analizu mreže se dalje mogu koristiti metrike i metode za analizu socijalnih mreža i omogućiti karakterizacija rezultata detekcije sličnosti u smislu različitih šema kolaboracije [1][3].

Izdvajaju se četiri karakteristične šeme kolaboracije među studentima uočene u postojećoj dugogodišnjoj praksi [1]:

- Izolovani par studentskih radova (u daljem tekstu: PAIR) čine dva programska koda sa izraženom detektovanom sličnošću, bez izraženih veza ka drugim radovima. Ova šema kolaboracije se jednostavno uočava i u izvornim rezultatima koje daju alati za detekciju sličnosti u programskom kodu.
- Grupa sličnih radova sa izraženim centralnim čvorom (u daljem tekstu: STAR). Tipično, u ovakvim situacijama jedan student predstavlja izvor plagijarizma, bilo slanjem programskog koda drugim kolegama, javnim objavljivanjem koda na neki drugi način ili promovisanjem izvora rešenja. Često je centralni čvor akter koji je izvor plagijarizma, ali ukoliko je u takvoj grupi sličnih radova jedna od veza između čvorova izraženija i uključuje centralni čvor, onda to može biti i čvor na „periferiji“. To je najčešće u

slučajevima kada su sam originalni izvor i distributer dve različite, ali intenzivno povezane osobe.

- Grupa sličnih radova bez izraženog centralnog čvora (u daljem tekstu: GROUP). Ovakva situacija često znači da izvor plagijarizma nije dostupan iz nekog razloga. Jedan od razloga može biti zajednički rad studenata, gde svi onda predaju to zajedničko rešenje ili rešenja sa zajedničkim delovima, kao i mogućnost upotrebe izvora sa interneta ili generativnih alata veštačke inteligencije. Takođe, javlja se i situacija u kojoj treće lice sa nadoknadom ili bez nje piše ili ustupa rešenja grupi studenata koji ih onda mogu do neke mere modifikovati i predati kao svoje delo. Uobičajeno, takvi radovi će biti povezani u klaster nalik kliku.
- Pojedinačan student pronalazi i kombinuje programski kod iz više izvora da bi došao do rešenja sopstvenog zadatka (u daljem tekstu: MIX), ali tipično bez značajnije izmene delova rešenja. Tipično, ovakvi čvorovi su relativno izdvojeni i nalaze se na periferiji svojih klastera sa tendencijom povezivanja sa čvorovima iz različitih klastera.

Predmet ovogodišnjeg projektnog zadatka je kvantitativna i kvalitativna analiza strukture i funkcionisanja mreža sličnosti u programskom kodu zasnovanih na rezultatima sličnosti dobijenih primenom odgovarajućih softverskih alata kao što su *Moss* [4] i *JPlag* [5]. U tom kontekstu, potrebno je analizirati modelovane mreže sličnosti sa fokusom na njihove karakteristike kao kolaborativnih i koautorskih mreža, prisutnih šema kolaboracije i prostiranja informacija, kao i ulogu pojedinaca unutar posmatranih mreža.

3.2. Skup podataka za analizu

Skup podataka za analizu je nastao na osnovu predatih studentskih rešenja dva domaća zadatka sa jednog od predmeta iz oblasti programiranja i algoritama i struktura podataka na nižim godinama studija u proteklih pet godina. Zbog potreba održavanja privatnosti neće biti navedeni tačan naziv predmeta, godina studija, oblast kojom se bavi, ni školska godina u okviru koje su predati radovi. Svi podaci o studentima su na odgovarajući način anonimizovani kako bi se zaštitio identitet učesnika.

Skup podataka se sastoji od 2 *Excel* datoteke *Results_1_anonymized.csv* i *Results_2_anonymized.csv* koje se odnose na rezultate detekcije sličnosti predatih domaćih zadataka studenata na dva različita domaća zadatka korišćenjem alata *Moss*. Skup studenata se u najvećoj meri preklapa, ali nije u potpunosti isti, odnosno mogu postojati studenti koji su predali samo jedan domaći zadatak.

Podaci se sastoje od sledećih kolona:

<code>acter1, acter1_percentage, acter2, acter2_percentage, lines_matched</code>
--

Značenje pojedinačnih polja je sledeće:

- `acterX` – jedinstveni identifikator studenta X,
- `acterX_percentage` – procenat sličnosti rada studenta X u odnosu na rad studenta Y u poređenom paru,
- `lines_matched` – apsolutni broj linija programskog koda koji se poklapa u radovima studenta X i studenta Y.

Takođe, poznate su i informacije o određenom broju ručno potvrđenih slučajeva plagijarizma koji se mogu koristiti za validaciju rezultata ili kao polazna tačka za formiranje novih. Podaci se nalaze u datoteci *Confirmed_plagiarism.csv* po sledećem formatu:

<code>assignment_no, acter1, acter2, ..., acterN</code>

Jedan red datoteke sadrži podatke o jednoj grupi sličnih radova studenata definisanih identifikatorima `acterX` na domaćem zadatku definisanim poljem `assignment_no`.

3.3. Modelovanje mreža

Sekundarni skup podataka iz ove analize koristiće se za modelovanje odgovarajućih socijalnih mreža. U zavisnosti od toga da li se projekat radi samostalno ili u paru, postoje dve različite varijante projektnog zadatka. Ako studenti rade projekat samostalno ili u paru za maksimalno 20 od 40 poena, potrebno je da se fokusiraju na modelovanje odvojenih kolaborativnih mreža sličnosti u programskom kodu *DZ1Net* i *DZ2Net* i izvrše analize na osnovu tih mreža (potpoglavlje „Modelovanje mreža *DZ1Net* i *DZ2Net*“). Sa druge strane, studenti koji rade projekat u paru i žele da osvoje maksimalnih 40 poena, treba da modeluju i analiziraju i agregirane mreže (*DZAggNet* i *DZAggNetT*), kao i da koriste različite pragove za filtriranje sličnosti (grana) u procesu agregacije kako bi dodatno istražili uticaj težine grana sa malom sličnošću na strukturu objedinjene mreže (potpoglavlje „Modelovanje mreža *DZAggNet*”).

Modelovanje mreža DZ1Net i DZ2Net

Studenti koji rade projekat samostalno, ili u paru za maksimalno 20 od 40 poena, potrebno je da se fokusiraju na modelovanje mreža sličnosti u programskom kodu *DZ1Net* i *DZ2Net*. Ove mreže

modelovati na osnovu dostupnih podataka, koristeći odgovarajuće tipove mreža (usmerena, neusmerena, težinska, itd.) u skladu sa postavljenim istraživačkim pitanjima i ciljevima analize.

U obe mreže, studenti koji su predali radove u okviru odgovarajućih domaćih zadataka predstavljaju čvorove, a veza između dva čvora se uspostavlja ukoliko postoji detektovana sličnost u odgovarajućem skupu rezultata detekcije sličnosti. Za analizu, potrebno je koristiti samo mreže *DZ1Net* i *DZ2Net*. Agregacija ovih mreža u zajedničku mrežu (*DZAggNet*) nije potrebna za ovaj nivo zadatka.

Modelovanje mreža *DZAggNet* i *DZAggNetT*

Studenti koji rade projekat u paru za maksimalnih 40 poena, potrebno je da agregacijom podataka o detekciji sličnosti na dva predata domaća zadatka. Suštinski, mreža *DZAggNet* se dobija linearnom kombinacijom mreža *DZ1Net* i *DZ2Net*:

$$w_{ij}^O = a \cdot w_{ij}^T + b \cdot w_{ij}^S$$

gde su w_{ij}^T i w_{ij}^S težine veza između čvorova i i j u mrežama *DZ1Net* i *DZ2Net*, dok su a i b koeficijenti ponderisanja koji odražavaju relativni značaj svake mreže u agregaciji. Potrebno je kreirati mrežu *DZAggNet* tako da dodatno istakne veze onih parova koji se javljaju u obe prethodno definisane mreže. Izbor koeficijenata a i b se ostavlja studentima koji rešavaju zadatak i predstavlja deo rešenja. Potrebno je kreirati mrežu *DZAggNetT* (*agregiranu mrežu u kojoj je dodatno primenjen i prag za minimalnu težinu grane koja se zadržava*) tako da se na osnovu mreže *DZAggNet* izvrši dodatno filtriranje grana male značajnosti na osnovu definisanog praga c . Izbor praga c za filtriranje grana male značajnosti se ostavlja studentima koji rešavaju zadatak i predstavlja deo rešenja.

Objedinjene mreže *DZAggNet* i *DZAggNetT* omogućavaju sveobuhvatnu analizu integriranjem svih informacija, pružajući širu perspektivu o obimu i strukturi interakcija unutar posmatranih kolaborativnih mreža sličnosti u programskom kodu. Pravilan izbor vrednosti težinskih koeficijenata će doprineti robusnosti zaključaka i identifikaciji ključnih obrazaca interakcija.

3.4. Istraživačka pitanja i ciljevi

Prilikom obrade primarnog i sekundarnog skupa podataka pogodno je kao smernice koristiti prethodno definisana istraživačka pitanja. U okviru ove sekcije je postavljen jedan broj takvih pitanja, a studenti treba da, nakon što odgovore na ova pitanja, na osnovu analize problema i samih podataka definišu dodatna pitanja ili specijalizuju navedena čime mogu bliže usmeriti samu analizu. Odgovore na pitanja treba dati u formi specificiranoj u poglavlju 4.

Pitanja su grupisana u kategorije po tematici koju obrađuju i tehnikama analize koje se u njima sprovode. Ako nije drugačije navedeno, svako od pitanja se odnosi na svaku od konstruisanih mreža, ukoliko je za nju smisleno i primenljivo. Studentima se predlaže da prvo sve analize sprovedu za jednu od mreža, pa zatim pređu na sledeću i da svoje odgovore na taj način strukturiraju i izlože u izveštaju o projektu.

Da bi se odgovorilo na postavljena pitanja, potrebno je primeniti odgovarajuće mere i metode za analizu mreže ili statističke metode. Mreže bi trebalo karakterisati kako kroz osnovna svojstva mreže, tako i kroz složenije mere centralnosti i metode za detekciju komuna. Mere i metode izabrati prema adekvatnosti spram postavljenog problema. Tamo gde se očekuje odgovor u obliku neke vrste rangiranja, navesti listu od 5 do 10 najrelevantnijih rezultata.

3.4.1. Osnovna karakterizacija modelovanih mreža (10/20 poena)

- 1) Kolika je gustina mreže?
- 2) Kolike su prosečne distance u okviru mreže i dijametar mreže?
- 3) U kojoj meri je mreža povezana i centralizovana? Navesti broj i veličine povezanih komponentata i proceniti da li postoji gigantska komponenta. Specifično komentarisati ulogu praga c u kontekstu mreže *DZAggNetT*. *Ovo je potrebno i matematički potkrepiti, recimo tako što bi se prikazala zavisnost broja komponentata u mreži DZAggNetT.*
- 4) Koliki je prosečni, a koliki globalni koeficijent klasterizacije mreže? Kakva je raspodela lokalnog koeficijenta klasterizacije njenih čvorova? Da li je klasterisanje izraženo ili ne? Odgovor dati upoređivanjem sa slučajno generisanim *Erdos-Renyi* i *scale free* mrežama istih dimenzija. Specifično komentarisati ulogu praga c u kontekstu mreže *DZAggNetT*.
- 5) Izračunavanjem relevantnih mrežnih metrika formalno obrazložiti da li mreža iskazuje osobine malog sveta.
- 6) Izvršiti asortativnu analizu po stepenu čvora i dati odgovor da li je i koliko izraženo asortativno mešanje. Obratiti pažnju da je stepen čvora numerička, a ne kategorička varijabla i izabrati adekvatnu metriku za meru asortativnog mešanja. Priložiti i vizuelizaciju.
- 7) Kakva je distribucija čvorova po stepenu i da li prati *power law* raspodelu? Za dobijanje punih poena za ovo pitanje potrebno je formalno matematički potkrepiti odgovor.

3.4.2. Analiza mera centralnosti (4/20 poena)

- 8) Sprovesti analize centralnosti po stepenu, bliskosti i relacionoj centralnosti. Dati pregled najvažnijih aktera po svakoj od njih.
- 9) Ko su najvažniji akteri po centralnosti po sopstvenom vektoru ili ekvivalentnim metrikama (*hub score*, *authority score*) ako je mreža modelovana kao usmereni graf? Šta nam to govori o njima?
- 10) Na osnovu prethodna dva pitanja predložiti i konstruisati heuristiku (kompozitnu meru centralnosti) za pronalaženje najvažnijih aktera i pronaći ih. Obratiti pažnju na tip mreže koji se analizira (usmerena ili neusmerena) i, shodno tome, prilagoditi koliko različite mrežne metrike utiču na heuristiku.

3.4.3. Detekcija komuna Luvenskom metodom (2/20 poena)

- 11) Sprovesti klasterisanje Luvenskom metodom (maksimizacijom modularnosti) u alatu *Gephi* za tri različite vrednosti parametra rezolucije. Konstruisati vizuelizacije i diskutovati izbor parametra rezolucije na dobijeno klasterisanje (broj i veličina klastera). Uzeti u obzir i poznate, potvrđene, slučajeve plagijarizma dostupne u primarnom skupu podataka. Na osnovu ove diskusije predložiti dobru vrednost parametra rezolucije.
- 12) Koje zajednice (komune) se mogu uočiti prilikom analize mreže, a koji akteri su ključni brokeri? Da li postoji neko objašnjenje za detektovane komune, recimo na osnovu četiri karakteristične šeme kolaboracije date u poglavlju 3.1?

3.4.4. Detekcija komuna spektralnim klasterisanjem (2/20 poena)

- 13) Sprovesti spektralnu analizu i proceniti potencijalne kandidate za broj komuna u mreži. Uporediti rezultat sa dendrogramom konstruisanim *Girvan-Newman* metodom.
- 14) Ko su akteri koji se mogu okarakterisati kao ključni brokeri (mostovi) u mreži? Šta ih čini brokerima? Porediti odgovor sa ključnim čvorovima dobijenim analizom centralnosti.

3.4.5. Poređenje modelovanih mreža (2/20 poena)

- 15) Sprovesti poređenje rezultata obrade modelovanih mreža u smislu definisanja glavnih aktera, karakterističnih obrazaca interakcije i sl. Specifično, uporediti rezultate sa četiri karakteristične šeme kolaboracije među studentima koje su zadate u poglavlju 3.1 i informacijama o ručno potvrđenim slučajevima plagijarizma definisanim u poglavlju 3.2.

3.5. Preporučene metode i alati

Za analizu modelirane socijalne mreže se preporučuje korišćenje programskih jezika *Python* (*NetworkX* biblioteka) i R (*sna* i *igraph* paketi) ili softverskih alata Gephi, UCINET, ili Pajek. Obrada primarnog skupa podataka se može obaviti pomoću MS *Excel* alata ili pisanjem odgovarajućih skripti u programskom jeziku po izboru. Ukoliko nije moguće drugačije, razrešavanje eventualnih dvosmislenosti u primarnom skupu podataka izvršiti ručno.

Vizuelizacija mreže se može obaviti korišćenjem alata Gephi, NodeXL, Cytoscape ili kroz podršku u okviru programskih jezika *Python* (*matplotlib*, *graphviz* i *graph-tool* biblioteke) i R (*igraph* paket).

4. REZULTATI

Projektni zadatak se predaje u vidu pisanog izveštaja koji sadrži rezultate sprovedene analize i pisana objašnjenja uočenih fenomena. Uz izveštaj se dostavljaju i odgovarajuće dopunske datoteke, kao što su tabele sa rezultatima analize, izvorni programski kod skripti ili programa korišćenih u analizi, datoteke koje sadrže produkovane vizuelizacije i sl. Potpuno odsustvo dopunskih datoteke koje predstavljaju rezultate rada može povući umanj enje broja poena na projektnom zadatku. Za pisanje izveštaja se može koristiti šablon koji se nalazi u odgovarajućoj sekciji na sajtu predmeta. Preporučeni obim izveštaja je do 10 stranica teksta.

5. PREDAJA, ODBRANA I VREDNOVANJE

Projektni zadatak se predaje elektronskim putem najkasnije do termina ispita u odgovarajućem ispitnom roku na način kako to bude specificirao predmetni nastavnik. Na odbranu je potrebno doneti štampanu verziju izveštaja. Po pravilu, projektni zadatak se brani pred predmetnim nastavnikom ili saradnikom u ispitnom roku u kome student želi da polaže ispit. Ukoliko student želi da brani zadatak u nekom drugom terminu, treba o tome da blagovremeno obavesti predmetnog nastavnika, radi eventualnog dogovora. Ukoliko se projektni zadatak radi u paru, studenti zajedno brane projektni zadatak.

Projektni zadatak nosi maksimalno 40 poena, a razlike u zahtevima između različitih verzija projekta su opisane u sekciji 3.3. Raspodela poena po tematskim oblastima je prikazana u sekciji 3.4. Studenti ne moraju realizovati sve zahteve u okviru projekta pre odbrane. Ne postoji minimalan broj poena koji je potrebno osvojiti na projektnom zadatku da bi se položio ispit.

Poeni sa jednom odbranjenog projektnog zadatka važe jednu školsku godinu. Postoji mogućnost da se dobro urađeni projektni zadaci prošire u završni, master rad. Upit u vezi sa takvom mogućnošću studenti mogu uputiti predmetnom nastavniku ili saradniku.

LITERATURA

- [1] M. Mišić, "Unapređenja sistema za detekciju plagijarizma u izvornom programskom kodu", Univerzitet u Beogradu – Elektrotehnički fakultet, doktorska disertacija, maj 2017.
- [2] M. Ahmadzadeh, E. Mahmoudabadi, and F. Khodadadi, "Pattern of plagiarism in novice students' generated programs: An experimental approach," *Journal of Information Technology Education*, vol. 10, 2011.
- [3] A. Zrnec and D. Lavbič, "Social network aided plagiarism detection," *British Journal of Educational Technology*, 2015.
- [4] S. Schleimer, D. S. Wilkerson, and A. Aiken, "Winnowing: local algorithms for document fingerprinting," in *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, 2003, pp. 76-85: ACM.
- [5] L. Prechelt, G. Malpohl, and M. Philippsen, "Finding plagiarisms among a set of programs with JPlag," *Journal of Universal Computer Science*, vol. 8, no. 11, pp. 1016-1038, 2002 2002.