SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 6667

ZAŠTITA KORISNIČKE PRIVATNOSTI METODOM ANONIMIZACIJE POVIJESTI PRETRAŽIVANJA WEB-PREGLEDNIKA

Mihaela Svetec

Zagreb, lipanj 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 6667

ZAŠTITA KORISNIČKE PRIVATNOSTI METODOM ANONIMIZACIJE POVIJESTI PRETRAŽIVANJA WEB-PREGLEDNIKA

Mihaela Svetec

Zagreb, lipanj 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 13. ožujka 2020.

ZAVRŠNI ZADATAK br. 6667

Pristupnica:	**Mihaela Svetec (0036500791)**
Studij:	Računarstvo
Modul:	Računarska znanost
Mentor:	doc. dr. sc. Klemo Vladimir

Zadatak: **Zaštita korisničke privatnosti metodom anonimizacije povijesti pretraživanja web-preglednika**

Opis zadatka:

Proučiti i opisati PBooster algoritam za zaštitu korisničke privatnosti na Internetu postupkom anonimizacije povijesti pretraživanja. Usporediti optimizacijsku metodu algoritma PBooster sa sličnim poznatim rješenjima i navesti njene prednosti i nedostatke. Osmisliti i implementirati generator sintetičkog skupa podataka prikladnog za provedbu postupka anonimizacije povijesti pretraživanja. Implementirati osnovnu inačicu algoritma PBooster, provesti eksperimentalno vrednovanje opisanog postupka te predstaviti i objasniti rezultate.

Rok za predaju rada: 12. lipnja 2020.

Sadržaj

Uvod

Ovisno o pojedincu, pojam resursa poprima različito značenje. To može biti vrijeme, novac, imovina, obrazovanje – bilo što, materijalno ili nematerijalno, što može proizvesti neku vrijednost ili je samo po sebi vrijedno. Resurs možemo zamisliti kao klasu čiji objekti zadovoljavaju spomenuti kriterij. Razvoj tehnologije i digitalizacija svakodnevnog života pridonijeli su stvaranju novog objekta te klase – osobni podaci. Osobni podaci postali su vrijedan resurs koji posjeduje svaki pojedinac te jedino on ima pravo odlučivati kako će se oni koristiti. Nažalost, čini se kako se u današnjem svijetu to pravo zanemaruje.

Prilikom korištenja Interneta, korisnik očekuje okruženje koje mu, uz traženo zadovoljstvo, pruža i određenu razinu privatnosti. Privatnost se općenito odnosi na slobodu od upada i slobodu odlučivanja hoće li se, kako i kome otkrivati nečiji osobni podaci. U okviru pretraživanja na Internetu, privatnost vežemo uz zaštitu korisničkih podataka i povijesti pretraživanja. Ako ih promatramo kao podskup osobnih podataka, oni postaju resurs nad kojim pravo upravljanja posjeduje samo korisnik. Povijest pretraživanja dio je digitalnog traga koji korisnik, svojom aktivnošću na mreži, ostavlja za sobom i gotovo ga je nemoguće izbrisati. Na korisniku preostaje da sam zaštiti svoju aktivnost na Internetu. Neka od mogućih rješenja su korištenje raznih web-ekstenzija za „zagađivanje“ povijesti pretraživanja, *VPN*-a (engl. *Virtual Private Network*), web-preglednika za anonimno pretraživanje itd. Navedene metode, iako pružaju prilično visoku razinu privatnosti, dovode do pada u kvaliteti pretraživanja. Dolazimo do pitanja: je li moguće očuvati kvalitetu pretraživanja uz rast privatnosti?

U nastavku je predstavljen *PBooster*, metoda anonimizacije povijesti pretraživanja koja uvodi optimalan omjer privatnosti i kvalitete. *PBooster* razvili su Ghazaleh Beigi et al. (2019.) te ga opisali u radu *Protecting User Privacy: An Approach for Untraceable Web Browsing History and Unambiguous User Profiles*. Naglasak ovog završnog rada stavljen je na glavne značajke i funkcionalnosti *PBoostera* te implementaciju osnovnih komponenti. Također, opisan je i proveden postupak analize osnovnih funkcija nad umjetno stvorenom strukturom podataka. Opisan je i model deanonimizacije na kojem se temeljio razvoj *PBoostera*. U konačnici, dana je usporedba algoritma sa sličnim rješenjima, uz navedene prednosti i nedostatke.

1. Postupak deanonimizacije

Kako bi metoda anonimizacije povijesti pretraživanja bila jasnija, poželjno je upoznati se s metodom deanonimizacije na temelju koje je ona stvorena. Postupak deanonimizacije povijesti pretraživanja predstavlja napad povezivanjem – povezuje povijest pretraživanja s krajnjim korisnikom.

1.1. Povijest pretraživanja

Povijest pretraživanja definirana je kao lista web-stranica koje je korisnik posjetio koristeći po želji odabrani web-preglednik. Svaka web-stranica opisana je pripadajućim imenom i *URL*-om. Web-preglednik pohranjuje povijest pretraživanja u obliku dnevnika na lokalni tvrdi disk. Osim preglednika, povijesti pretraživanja mogu pristupati tzv. *third-party trackers* (lokatori prisutni na web-stranicama koji, osim samih stranica, bilježe posjet korisnika) te *ISP*-ovi (engl. *Internet Service Provider*). Zbog velike izloženosti, povijest pretraživanja postaje glavna meta napada na korisnikovu privatnost. Industrija internetskog oglašavanja koristi spomenute lokatore treće strane kako bi izgradila povijesti pretraživanja pojedinaca. Većina njenih pripadnika tvrdi kako se povijesti pretraživanja pohranjuju pod pseudonimima i nikako ne povezuju s identitetom krajnjeg korisnika. Međutim, jedinstvenost povijesti pretraživanja dovodi do pitanja: može li skup web-poveznica identificirati krajnjeg korisnika?

1.2. Model prijetnje

Neprestani razvoj tehnologije pridonio je integraciji društvenih mreža u sve aspekte života, stvarajući pritom digitalnu platformu čija namjena je svestrana i ubrzana razmjena podataka. Kreiranjem profila na društvenoj mreži korisnik postaje dijelom te platforme. Profil društvene mreže stvara se na temelju osobnih podatka i postaje dijelom korisnikovog digitalnog traga.

U radu *De-anonymizing Web Browsing Data with Social Networks* [2] predstavljen je postupak deanonimizacije povijesti pretraživanja koji, na temelju javno dostupnih podataka, povezuje povijest pretraživanja s profilom na društvenim mrežama. Ideja deanonimizacijskog modela proizlazi iz sklonosti korisnika posjećivanju poveznica koje se

nalaze na naslovnim stranicama društvenih mreža. Dodatno, svaki korisnik ima karakterističan skup prijatelja i vjerojatnije je da će spomenute poveznice pripadati objavama nekog iz skupa prijatelja. Kako odabrane poveznice postaju dijelom povijesti pretraživanja, dolazi do stvaranja prepoznatljivih obrazaca pregledavanja web-stranica. Konačno, pretpostavlja se da je poznat podskup povijesti pretraživanja kojem pripadaju samo poveznice odabrane koristeći društvene mreže. Primjenom metode maksimalne vjerodostojnosti uz navedene pretpostavke, model prijetnje identificira profil društvene mreže koji je najvjerojatnije generirao promatranu povijest pretraživanja. Napad deanonimizacije može se formalno prikazati na sljedeći način:

„Koristeći povijest pretraživanja korisnika *u,* $H^u = \{l_1, ..., l_n\}$, gdje l_i , $i \in [1, n]$ predstavlja poveznice od kojih se povijest pretraživanja sastoji, povezati korisnika *u* s profilom društvene mreže koji je najvjerojatnije generirao promatranu povijest pretraživanja.“ [2]

Vrednovanje predstavljenog deanonimizacijskog modela provedeno je nad sintetički generiranim skupom podataka te nad stvarnim podacima koji su prikupljeni u *online* istraživanju. Koristeći sintetički generirane podatke, model postiže uspješnost veću od 50%. Prilikom vrednovanja nad stvarnim podacima (korištena je društvena mreža *Twitter*), postiže se uspješnost modela veća čak od 70%. Zbog poprilično visokog postotka uspješnosti deanonimizacijskog modela, javlja se interes za razvojem protumodela koji bi se borio protiv deanonimizacije i štitio privatnost korisnika.

2. Postavljanje problema

Anonimizatorom se želi postići smanjenje učinka modela prijetnje opisanog u poglavlju 1. Cilj modela je anonimizacija povijesti pretraživanja korisnika tako da se, prema određenom kriteriju, izabere skup poveznica koji se zatim dodaje u početnu povijest pretraživanja. Ovakva metoda anonimizacije naziva se metodom „zagađivanja“ (povijest pretraživanja se maskira viškom podataka). Prilikom dodavanja poveznica potrebno je obratiti pozornost na dvije značajke: privatnost i korisnost. Pojam korisnosti u kontekstu anonimizacije povijesti pretraživanja bit će definiran kao razina zadovoljstva korisnika pružanom uslugom. Privatnost je sačuvana kada model prijetnje ne može povezati promatranu povijest pretraživanja s korisnikom. Ako je cilj maksimizirati privatnost, tada bi bilo dovoljno u povijest pretraživanja ubacivati nasumične poveznice dok se ne postigne tražena razina privatnosti. Međutim, treba primijetiti kako se tada smanjuje korisnost – pružana usluga postat će previše općenita, tj. karakteristike koje definiraju korisnika postat će zanemarive i usluga neće biti dovoljno personalizirana. Odnos privatnosti i korisnosti je obrnuto-proporcionalan. Kako bi model anonimizacije bio učinkovit u svakom smislu, potrebno je pronaći optimalan odnos između spomenutih značajki.

2.1. Struktura podataka

Prvi korak u implementaciji svakog modela je organizacija podataka. Povijest pretraživanja osnovan je skup podataka za koje je potrebno utvrditi prihvatljivu strukturu. Poveznice opisuju web-stranice, što intuitivno dovodi do ideje grupiranja (engl. *clustering*) poveznica prema sadržaju stranica. Jedan od načina grupiranja bio bi unaprijed definirati kategorije te na temelju sadržaja web-stranice smjestiti njenu poveznicu u pripadnu kategoriju. Svaka kategorija bila bi opisana ključnim riječima, a podjela u kategorije temeljila bi se na izračunu sličnosti tekstualnog sadržaja web-stranice s ključnim riječima. Na ovaj način svaki korisnik bi bio predstavljen skupom kategorija i udjelom kategorija u povijesti pretraživanja. Navedeno rješenje nije prikladno iz nekoliko razloga. Ključne riječi kategorija bilo bi potrebno često ažurirati, što bi s porastom količine podataka postalo poprilično neučinkovito. Također, grupiranje pruža grubu kategorizaciju sadržaja web-stranica što često može dovesti do pogreške zbog nepreciznosti.

Prikladniji način podjele poveznica je pomoću metode modeliranja tema. Modeliranje tema je vrsta statističkog modeliranja za otkrivanje apstraktnih tema koje se pojavljuju u skupu dokumenata. Teme su opisane skupom riječi, pri čemu je svakoj riječi pridijeljena vjerojatnost pojavljivanja u promatranoj temi. Različite teme mogu dijeliti pojedine riječi, a dokumenti mogu imati više tema. Web-stranice mogu se promatrati kao dokumenti, gdje sadržaj dokumenta odgovara tekstualnom sadržaju web-stranice. Nakon provedbe tehnike modeliranja nad korpusom (skupom dokumenata), dobivena je struktura tema s odgovarajućom distribucijom riječi. Na temelju dobivene strukture, za svaku web-stranicu utvrđuje se razina prisutnosti pojedine teme. Za reprezentativnu temu web-stranice uzima se tema s najvišom učestalosti.

Budući da tehnika modeliranja tema pripada nenadziranom strojnom učenju, ona ostvaruje brzu i jednostavnu analizu podataka. Najpoznatiji pristup modeliranju tema je tehnika *LDA* (engl. *Latent Dirchlet Allocation*). Za izgradnju strukture podataka na temelju povijesti pretraživanja, Beigi et al. [1] koriste *LDA* te je postupak u nastavku opisan.

2.2. Modeliranje tema – LDA

LDA je tehnika modeliranja tema koja generira teme na temelju učestalosti riječi iz skupa dokumenata. Dokumenti su predstavljeni kao slučajne kombinacije tema, a svaka tema karakterizirana je distribucijom riječi. U vektorskom prostoru, svaki korpus može se prikazati matricom dokument-riječ. Matrica je prikazana na slici 2.1.

	W1	W2	W3	Wm
D1	0	2	1	3
D2	1	4	0	0
D3	0	2	3	1
Dn	1	1	3	0

Slika 2.1 Matrica dokument-riječ

	W1	W2	W3	Wm
K1	0	1	1	1
K2	1	1	1	0
K3	1	0	0	1
K	1	1	0	0

Slika 2.2 Matrica tema-riječ

	K1	K2	K3	K
D1	1	0	0	1
D2	1	1	0	0
D3	1	0	0	1
Dn	1	0	1	0

Slika 2.3 Matrica dokument-tema

Matrica dokument-riječ prikazuje skup dokumenata veličine n (D_1, ..., D_n) i vokabular veličine m riječi (W_1, ..., W_m). Vrijednost ćelije retka i i stupca j označava broj pojavljivanja riječi W_j u dokumentu D_i. Matričnom faktorizacijom *LDA* tehnika pretvara matricu

dokument-riječ u dvije matrice: matricu tema-riječ i matricu dokument-tema. Matrica tema-riječ prikazana je na slici 2.2 – dimenzije matrice su (*K*, *m*), gdje je *K* veličina skupa tema. Matrica dokument-tema dimenzija je (*n*, *K*) i prikazana je na slici 2.3. Obje matrice sadrže vrijednosti traženih distribucija. *LDA* tehnikom uzorkovanja unaprijeđuje obje matrice, čime se na kraju dobiva stanje u kojem su distribucije preciznije.

Pridjeljivanje tema web-stranicama odvija se u nekoliko koraka. U početku se iz povijesti pretraživanja konstruira korpus nad kojim se primjenjuje *LDA* model. Na temelju tekstualnog sadržaja web-stranica i dobivene strukture tema u prethodnom koraku, određuju se učestalosti tema na svakoj od stranica. Naposljetku, najučestalija tema odabire se kao reprezentativna tema stranice. Dobivena struktura naučenih tema označava se izrazom $\mathrm{T} = \{t_1, \dots, t_m\}$. Svaka poveznica iz povijesti pretraživanja $H^u = \{l_1, \dots, l_n\}$ može se povezati s temom iz skupa T čime se dobiva matrica $T^u \epsilon \mathbb{R}^{n \times m}$. Matrica T^u predstavlja matricu odnosa poveznica-tema, gdje $T^u_{ij} = 1$ indicira da je poveznica *i* iz H^u u korelaciji s temom *j* iz T.

Konačno, anonimizacija povijesti pretraživanja korisnika *u* može se formalno definirati na sljedeći način:

„Koristeći povijest pretraživanja H^u korisnika *u* i matricu odnosa poveznica-tema T^u, anonimizacijski model f koristi se za stvaranje manipulirane povijesti pretraživanja $\widetilde{H^u}$ tako što se poveznice dodaju u H^u radi očuvanja privatnosti, uz zadržavanje korisnosti od $\widetilde{H^u}$ za buduće primjene.“ [1]

$$f: \{H^u, T^u\} \rightarrow \{\widetilde{H^u}\} \qquad (1)$$

U sljedećem poglavlju opisan je model iz [1] koji koristi definiranu strukturu podataka za postupak anonimizacije povijest pretraživanja korisnika.

3. Prijedlog rješenja – *PBooster*

Algoritam *PBooster* osnovna je komponenta anonimizacijskog modela koji su predložili Beigi et al. [1]. *PBooster* usmjeren je na optimizaciju privatnosti i korisnosti – značajke se promatraju u međusobnom odnosu, a ne neovisno. Kako bi mogli opisati rezultat i kvalitetu algoritma, uvode se mjere za izračun učinka dodavanja poveznica na privatnost i korisnost.

3.1. Mjerenje privatnosti

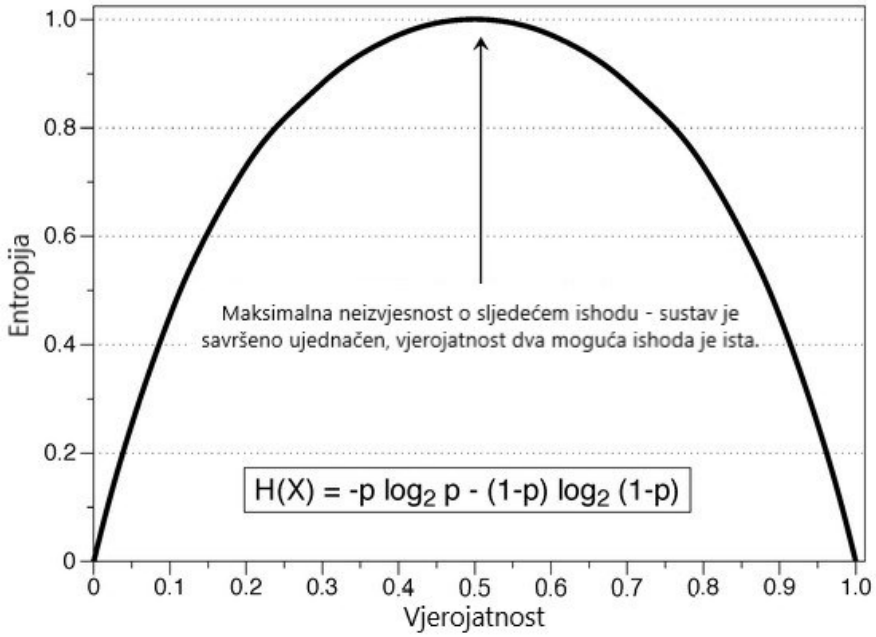
Svaki korisnik generira jedinstvenu povijest pretraživanja. Također, već je spomenuto kako poveznice modeliraju jedinstvene obrasce pregledavanja web-sadržaja. Obrasci pregledavanja vežu se uz korisnike čime se svakom korisniku pripisuju određene karakteristike. Opiše li se povijest pretraživanja korisnika matricom T^u iz prethodnog poglavlja, tada se karakteristike korisnika mogu prikazati distribucijom poveznica po temama. Zbog raznolikosti u interesu korisnika, jasno je kako spomenuta distribucija neće biti jednolika. Nejednolikost je upravo ono što korisnika čini jedinstvenim u grupi korisnika. Dolazimo do pitanja: što bi se dogodilo kada bi distribucija poveznica po temama bila jednolika? Jednolikost distribucije povećava dvoznačnost interesa korisnika te ga je teže izdvojiti iz grupe. U kontekstu anonimizacije, zaključuje se kako jednolikost distribucije poveznica po temama dovodi do povećanja anonimnosti, tj. očuvanja privatnosti korisnika.

Za korisnika *u*, distribucija poveznica po temama će biti opisana vektorom $c_u \in \mathbb{R}^{m\times 1}$, $c_u = (c_{u1}, ..., c_{um})$, gdje je *m* veličina skupa T . Vrijednost $c_{uj}, j \in [1, \dots, m]$, jednaka je broju poveznica u povijesti pretraživanja H^u korisnika *u* koje su vezane uz temu $t_j \in$ T. Kako vrijedi da je $\sum_{j=1}^{m} c_{uj}$ jednaka veličini povijesti pretraživanja (ukupnom broju poveznica, $|H^u|$), definira se vektor $p_u = J(c_u) = (p_{u1}, ..., p_{um})$, $p_u \in \mathbb{R}^{m\times 1}$, gdje $J(c_u)$ predstavlja funkciju normalizacije koja se provodi nad vektorom c_u. Vrijednost p_{uj}, $j \in [1, \dots, m]$, odgovara vrijednosti kvocijenta $\frac{c_{uj}}{|H^u|}$. Drugim riječima, vektor p_u predstavlja distribuciju vjerojatnosti tema. Ako c_u promatramo kao diskretnu slučajnu varijablu, tada p_u odgovara njenoj distribuciji $\begin{pmatrix} t_1 & \dots & t_m \\ p_{u1} & \dots & p_{um} \end{pmatrix}$. Za izračun privatnosti korisnika koristit će se entropija diskretne slučajne varijable. Entropija se računa prema izrazu (2). Varijabla *X*

predstavlja diskretnu slučajnu varijablu koja poprima vrijednosti iz skupa $\{x_1, \dots, x_n\}$, a $p(x_i)$ je vjerojatnost pojavljivanja x_i, i $\in [1, \dots, n]$.

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i) \tag{2}$$

Kada je distribucija diskretne slučajne varijable jednolika, tada entropija postiže maksimalnu vrijednost i ona iznosi $\log_{\boldsymbol{b}} n$. Ovo svojstvo lako je prikazati pomoću grafa funkcije entropije. Na slici 3.1 [8] dan je grafički prikaz funkcije za varijablu s dvije moguće vrijednosti. Kako su moguća samo dva ishoda, njihove vjerojatnost su jednake i iznose 0.5. Funkcija entropije postiže maksimum za vjerojatnost 0.5 – jednolika razdioba rezultira maksimalnom entropijom.



Slika 3.1 Grafički prikaz entropije diskretne slučajne varijable s dva moguća ishoda

Kako anonimizacija povijesti pretraživanja teži jednolikosti distribucije vjerojatnosti tema, entropija se uzima kao prikladno rješenje za definiranje mjere privatnosti. Sukladno izrazu za (2), privatnost se definira izrazom (3). Što je rezultat izraza veći, veća je i privatnost korisnika *u*.

$$Privatnost(c_u) = -\sum_{j=1}^{m} p_{uj} \log p_{uj} \tag{3}$$

3.2. Mjerenje korisnosti

Korisnost opisuje razinu zadovoljstva korisnika uslugom pružanom tijekom pregledavanja web-sadržaja. Svakom korisniku pruža se personalizirana usluga izgrađena na temelju prošlih *online* aktivnosti. Kako se te aktivnosti pohranjuju u povijest pretraživanja, nakon dodavanja novih poveznica u svrhu anonimizacije, potrebno je u obzir uzeti utjecaj te promjene na korisnost. Ako se distribucija vjerojatnosti tema nakon manipuliranja poviješću pretraživanja označi s $\widetilde{p_u}$, tada se promjena korisnosti može dobiti usporedbom p_u i $\widetilde{p_u}$. Beigi et al. [1] predstavljaju gubitak korisnosti kao razliku između navedenih distribucija, a za izračun uvode izraz (4).

$$korisnost_{gubitak}(p_u, \widetilde{p_u}) = 0.5 * (1 - sim(p_u, \widetilde{p_u})) \tag{4}$$

Sim predstavlja funkciju koja se koristi za izračun sličnosti dviju distribucija. Distribucije vjerojatnosti tema su prikazane kao n-dimenzionalni vektori, pa se za utvrđivanje sličnosti među njima primjenjuje sličnost kosinusa.

$$\cos\alpha = \frac{v_1 \times v_2}{\|v_1\|\|v_2\|} \tag{5}$$

Sličnost kosinusa česta je metoda koja se primjenjuje za izračun sličnosti dokumenata. Dokumenti se opisuju n-dimenzionalnim vekorima čije vrijednosti odgovaraju učestalosti pojedinih riječi u dokumentu. Između vektora v_1 i v_2 se računa kosinus kuta prema izrazu (5). Što je kut manji, sličnost među vektorima je veća, tj. dokumenti su slični. Sukladno općenitom izrazu, nad distribucijama p_u i $\widetilde{p_u}$ primjenjuje se sličnost kosinusa prema izrazu (6).

$$sim(p_u, \widetilde{p_u}) = \frac{p_u \times \widetilde{p_u}}{\|p_u\|\|\widetilde{p_u}\|} \tag{6}$$

Kako vrijedi *sim* ∈[-1,1], gubitak korisnosti će biti u intervalu [0, 1]. Također, minimalna vrijednost gubitka dobiva se kada je $p_u = \widetilde{p_u}$ – dodavanjem poveznica nije došlo do promjene u distribuciji vjerojatnosti tema, što znači da gubitka korisnosti nema. Maksimalna vrijednost dobiva se kada su p_u i $\widetilde{p_u}$ različiti za svaku vrijednost distribucije različitu od nula – dodavanje poveznica potpuno je promijenilo distribuciju i gubitak korisnosti je velik.

4. Implementacija osnovnih komponenti

Distribucija vjerojatnosti tema izvedena iz povijesti pretraživanja korisnika opisuje njegove interese. Maksimalna privatnost postiže se kada su interesi korisnika jednoliko distribuirani po različitim temama. Jednolikost dovodi do povećanja anonimnosti generalizacijom karakteristika korisnika, čime ga je teže izdvojiti iz skupine. Svrha anonimizacijskog modela „zagađivanjem" je dodavanje poveznica u povijest pretraživanja do trenutka kada je postignuta tražena razina privatnosti. Moguće rješenje je dodavanje skupa nasumičnih poveznica – njihov web-sadržaj ne mora pripadati nijednoj od tema koje opisuju povijest pretraživanja. Međutim, navedeni pristup nije povoljan jer dovodi do pada korisnosti – pružana usluga neće biti dovoljno personalizirana. Potrebno je pronaći ravnotežu u odnosu ovih dviju značajki. Beigi et al. [1] predstavljaju *PBooster* algoritam koji rješava opisani optimizacijski problem uvođenjem izraza (7).

$$G(J(c_u), J(\widetilde{c_u}), \lambda) = \lambda * Privatnost\big(J(\widetilde{c_u})\big) - korisnost_{gubitak}\big(J(c_u), J(\widetilde{c_u})\big) \quad (7)$$

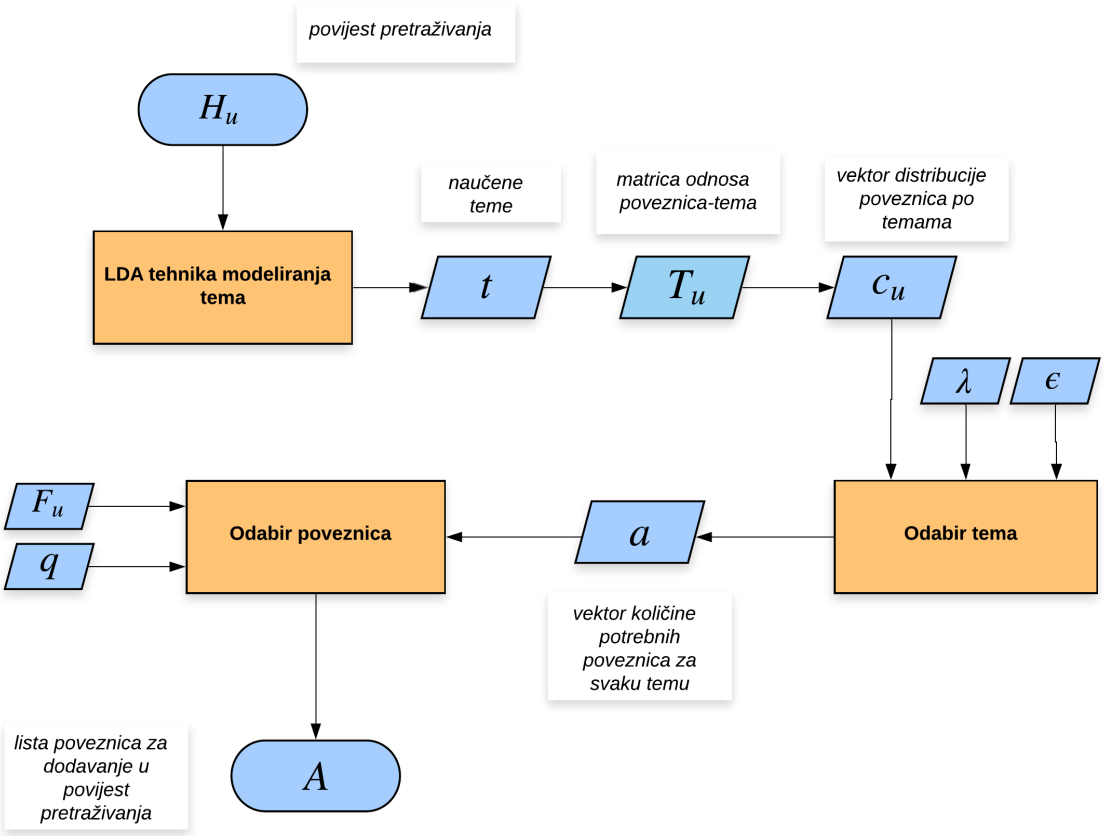
Funkcija *G* definira kvantifikaciju odnosa privatnosti i korisnosti. Svrha *PBooster* algoritma je pronaći točku maksimuma funkcije G – maksimumom su opisane optimalne vrijednosti značajki za koje anonimizacijski model postiže traženi učinak. Distribucija poveznica po temama, nakon manipuliranja poviješću pretraživanja, označena je vektorom $\widetilde{c_u}$. Parametar λ je upravljački parametar čija svrha je upravljanje doprinosom privatnosti. Što je vrijednost parametra λ veća, anonimnost povijesti pretraživanja će biti veća. Međutim, gubitak korisnosti će biti manji ako je parametar λ manji. Konačno, ako skup poveznica koje je potrebno dodati označimo s A, tada se problem pronalaska skupa može opisati na sljedeći način:

$$\mathrm{A}^* = \underset{\mathrm{A}}{\operatorname{argmax}}\, G(J(c_u), J(\widetilde{c_u}), \lambda) \quad (8)$$

Prostor pretraživanja optimizacijskog problema (8) raste eksponencijalno s brojem poveznica, čime potraga za optimalnim rješenjem postaje vrlo zahtjevna. Anonimizacijski model koji su predstavili Beigi et al. [1] koristi pristup problemu razlaganjem na dva potproblema. Prvi korak (odabir tema) usmjeren je na potragu za brojem poveznica koje je potrebno dodati za svaku od tema, s ciljem maksimizacije funkcije *G*. Rezultat prvog koraka je vektor $a = (a_1, \ldots, a_m) \in \mathbb{R}^{m\times 1}$, gdje $a_j, j \in [1, \ldots, m]$, odgovara broju poveznica koje je

potrebno dodati u povijest pretraživanja za temu $t_j \in T$. U drugom koraku (odabir poveznica), koristeći vektor a, odabire se skup prikladnih poveznica za svaku od tema.

Na slici 4.1, u obliku dijagrama toka podataka, prikazan je anonimizacijski model koji se temelji na *PBooster* algoritmu. Ulaz modela predstavlja povijest pretraživanja (skup poveznica, H^u) nad kojom se provodi tehnika *LDA* opisana u poglavlju 2. Rezultat modeliranja je skup tema koji se koristi za izgradnju matrice T^u. Vektor c_u proizlazi iz T^u te se nad njim provode optimizacijski koraci *PBooster* algoritma – odabir tema i odabir poveznica. Na izlazu modela pojavljuje se lista poveznica A koje je potrebno dodati u povijest pretraživanja. U nastavku poglavlja dan je detaljan opis oba optimizacijska koraka te ostvarenog programskog rješenja odabira tema.



Slika 4.1 Dijagram toka podataka modela koji se temelji na *PBooster* algoritmu

4.1. Odabir tema

Rješavanje optimizacijskog problema opisanog izrazom (8) može se provesti iscrpnim pretraživanjem kojim se, za svaku moguću vrijednost distribucije poveznica, provjerava optimalnost. Opisani pristup nije prihvatljiv zbog izrazite vremenske i prostorne složenosti problema. Proces optimizacije istovremeno mora pružati prihvatljivo rješenje i učinkovitost u kontekstu vremenske izvedbe.

Povijest pretraživanja predstavlja velik skup podataka te je moguće da, zbog količine informacija, dodavanje novih poveznica nema nikakav učinak na privatnost i korisnost. Anonimizacijski model donosi odluku o broju poveznica koje je potrebno dodati te je potrebno prilagoditi veličinu podataka povijesti pretraživanja kako bi anonimizacija bila uspješna. Beigi et al. [1] uvode pojam submodularnosti kojim se opisani problem može predočiti.

4.1.1. Nemonotone submodularne funkcije

Submodularnost je svojstvo funkcija koje se izvode nad skupovima podataka. Ako X predstavlja neki konačan skup te vrijedi $A \subseteq B \subseteq X$ i $e \in X \backslash B$, tada je funkcija $f: 2^X \rightarrow \mathbb{R}$ submodularna i vrijedi:

$$f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B) \tag{9}$$

Svojstvo submodularnosti pokazuje kako dodavanje elementa e skupu A dovodi do većeg povećanja u funkciji f u usporedbi s povećanjem koje se dobiva kada se element pridružuje skupu B, čiji je A podskup. Na temelju ovog svojstva izgrađuje se prva faza *PBooster* algoritma – odabir tema.

Izraz (7) sastoji se od dvije komponente – mjere privatnosti i mjere gubitka korisnosti. Mjera privatnosti opisana je funkcijom entropije koja je submodularna za nasumične vrijednosti varijable. Također, submodularna je i funkcija koja opisuje mjeru gubitka korisnosti. Funkcija G je kombinacija dviju submodularnih funkcija, što znači da je i ona sama submodularna. Dodatno, G je nemonotona. Ako izraz (8) zapišemo u obliku (10), vidljivo je da se problem optimizacije odnosa privatnosti i gubitka korisnosti, u svrhu anonimizacije povijesti pretraživanja, svodi na pronalazak maksimuma nemonotone submodularne funkcije.

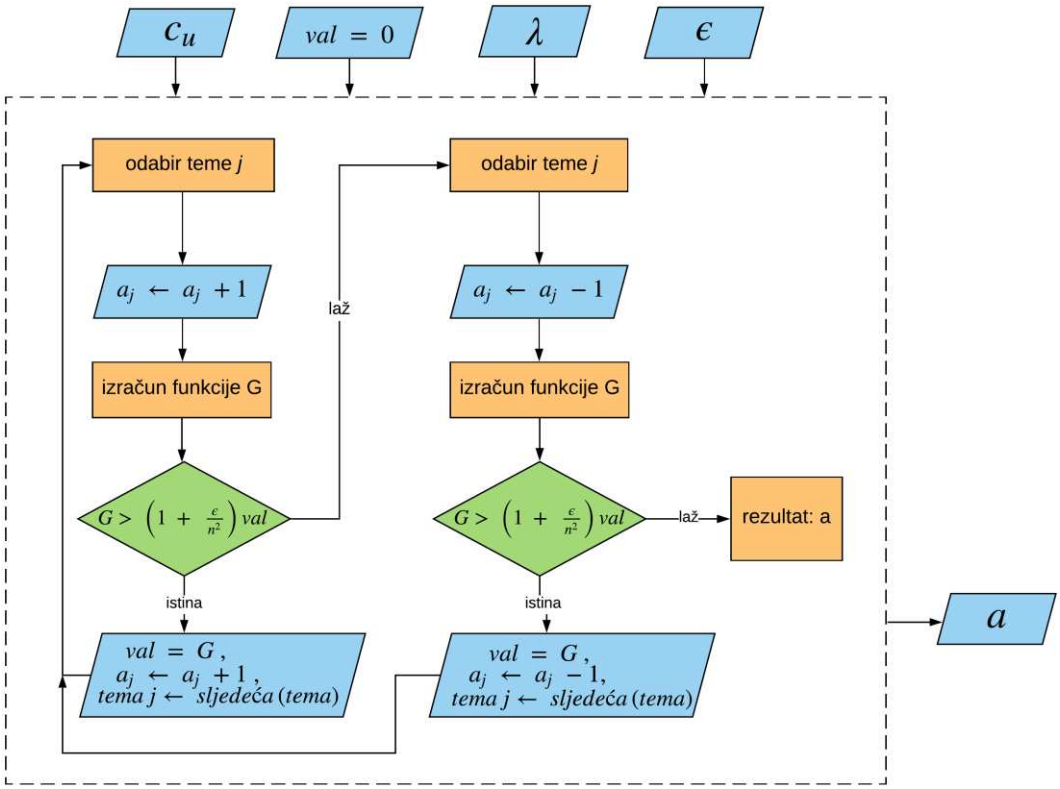
$$\underset{a}{\operatorname{argmax}}\left(-\lambda \sum_{j=1}^{m} p_{uj} \log p_{uj} - 0.5 * \left(1 - \frac{\sum_{j=1}^{m} p_{uj}\widetilde{p_{uj}}}{\sqrt{\sum_{j=1}^{m}\left(p_{uj}\right)^2}\sqrt{\sum_{j=1}^{m}(\widetilde{p_{uj}})^2}}\right)\right) \quad (10)$$

4.1.2. Pohlepni algoritam lokalne pretrage

Pronalazak maksimuma nemonotone submodularne funkcije je NP-težak problem te se ne može učinkovito izvesti u konačnom vremenu. Često korišten pristup opisanom problemu je uporaba pohlepnog algoritma lokalne pretrage (engl. *greedy local search algorithm*). Glavno obilježje pohlepnog algoritma je što u svakom koraku potrage uzima lokalni optimum, nadajući se da će u konačnici doći do globalnog optimuma. Algoritam se naziva pohlepnim jer u danom trenutku odabire rješenje koje se čini najboljim. Potraga za optimalnim rješenjem temelji se na uzastopnom dodavanju ili uklanjanju elemenata iz skupa s ciljem porasta vrijednosti submodularne funkcije. Kvalitete algoritma su jednostavnost i praktičnost, ali često može doći do zaustavljanja potrage u točkama lokalnih maksimuma. Prema [3], potraga za optimumom nemonotone submodularne funkcije primjenom pohlepnog algoritma rezultira rješenjem s vrijednosti najmanje $\frac{1}{3}$ optimalnog rješenja.

$$G(J(c_u), J(\widetilde{c_u}), \lambda) \geq (\tfrac{1}{3} - \tfrac{\varepsilon}{n}) G(J(c_u), OPT(c_u), \lambda) \text{ [3]} \quad (11)$$

Faza odabira tema implementira se primjenom pohlepnog algoritma nad vektorom a. Slika 4.2 grafički prikazuje tijek izvođenja algoritma. Parametar λ upravlja utjecajem privatnosti, a c_u opisuje distribuciju poveznica po temama za povijest pretraživanja korisnika u. Algoritam kao ulazni parametar prima i $\varepsilon \in [0,1]$ koji proizlazi iz izraza (11) ($OPT(c_u)$ označava optimalno rješenje). Početna vrijednost svih elemenata vektora a je nula. Nad elementima vektora a provodi se uvećanje ili smanjenje vrijednosti za 1 sve dok postupak rezultira rastom funkcije G. Vrijednosti funkcije G su u porastu sve dok je novoizračunata vrijednost veća od prethodne najmanje $(1 + \frac{\varepsilon}{n^2})$ puta [3]. U trenutku kada nije moguće postići daljnji porast, potraga se prekida te se trenutna vrijednost vektora a predstavlja kao optimalno rješenje.



Slika 4.2 Pohlepni algoritam za odabir tema

4.1.3. Programsko rješenje

Za implementaciju faze odabira tema korišten je programski jezik *Python*. Osnovnu strukturu podataka čine dvije klase: *PBooster* i *User*. Klasom *PBooster* stvara se objekt koji predstavlja anonimizacijski model. Objekt klase *PBooster* opisan je atributima *epsilon* i *lambda* – atributi predstavljaju statičke parametre anonimizacijskog modela, λ i ε. Klasa *User* koristi se za stvaranje objekata korisnika koji su opisani atributom *topic_frequency* – distribucijom poveznica po temama povijesti pretraživanja. Za pohranu distribucije poveznica po temama (*topic_frequency*) te broja novih poveznica (*to_be_added*) koristi se struktura rječnika. Ključevi rječnika predstavljaju oznake tema, a pohranjuju vrijednosti distribucija.

Funkcionalnost pohlepnog algoritma implementirana je funkcijom *topicSelection*. Funkcija kao argument prima objekt klase *User*, a kao rezultat vraća rječnik *to_be_added*. Izračun funkcije *G* provodi se pozivom funkcije *calculateFunctionG* koja kao argument prima *topic_frequency* i *to_be_added*. Funkcija *calculatePrivacy* koristi se za izračun mjere

privatnosti – kao argument prima rječnik *topic_frequency_new* dobiven iz *topic_frequency* i *to_be_added* zbrojem vrijednosti jednakih ključeva. Za izračun mjere gubitka korisnosti koristi se funkcija *calculateUtilityLoss* koja kao argumente prima *topic_frequency* te *topic_frequency_new*.

4.1.4. Pregled i vrednovanje rezultata

Vrednovanje implementiranog programskog rješenja provedeno je nad skupom umjetno stvorenih podataka. Podaci su pohranjeni u obliku tekstualne datoteke te je implementirana funkcija za učitavanje podataka.

Kako optimizacijska funkcija ovisi o parametru λ, vrednovanje rezultata provodi se nad vrijednostima iz skupa {0.5, 1, 5, 10, 15, 30, 50, 70, 100}. Parametru ε pridijeljena je vrijednost 0.01. Podaci o korisnicima prikazani su tablicom 4.1. Svaki korisnik opisan je veličinom povijesti pretraživanja i pripadnim vektorom distribucije poveznice po temama.

Tablica 4.1 Karakteristike povijesti pretraživanja korisnika

korisnik	$\|H^u\|$	c_u
korisnik1	10	< 2, 6, 2 >
korisnik2	15	< 3, 7, 1, 4 >
korisnik3	25	< 4, 5, 11, 2, 3 >
korisnik4	25	< 4, 8, 2, 1, 10 >
korisnik5	25	< 14, 1, 6, 4 >
korisnik6	25	< 7, 4, 3, 6, 1, 4 >
korisnik7	25	< 3, 2, 20 >
korisnik8	25	< 14, 11 >
korisnik9	25	< 1, 2, 3, 19 >
korisnik10	50	< 10, 3, 2, 15, 9, 11 >

korisnik11	70	< 9, 27, 5, 16, 8, 4, 1 >
korisnik12	100	< 13, 6, 15, 22, 17, 9, 11, 4, 3>
korisnik13	500	< 56, 3, 10, 60, 17, 29, 114, 79, 62, 24, 18, 28 >

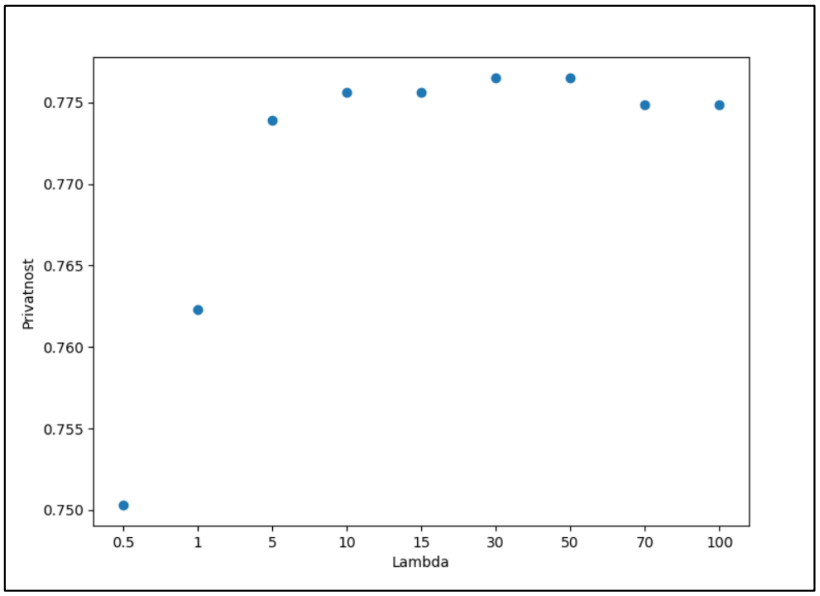
Tablica 4.2 prikazuje rezultate algoritma provedenog nad korisnikom *korisnik10*, $|H^u| = 50$, za različite vrijednosti parametra λ. Ovisnost privatnosti o λ prikazan je grafički slikom 4.3, a ovisnost korisnosti slikom 4.4 (korisnost je jednaka $1 - korisnost_{gubitak}$).

Tablica 4.2 Rezultati algoritma za korisnika *korisnik10* s promjenom u λ

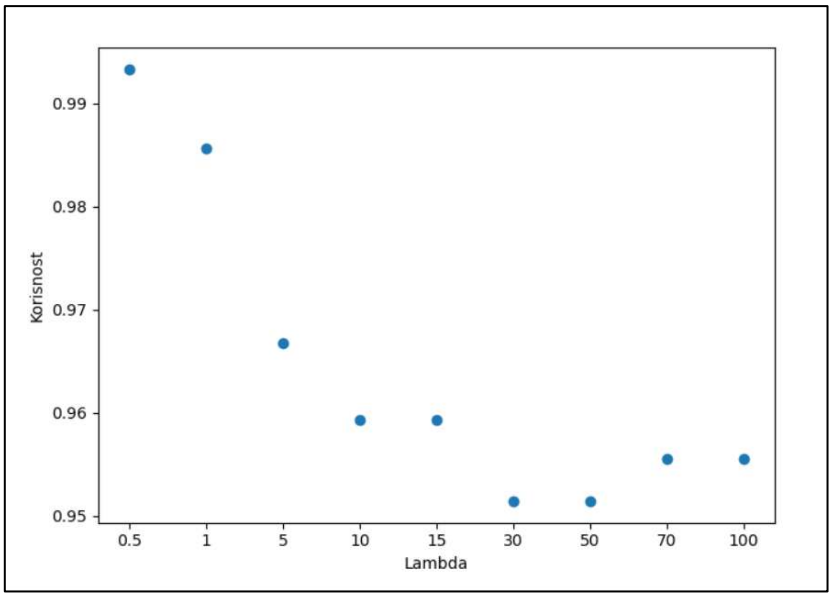
λ	**privatnost**	**korisnost**
0.5	0.7503330133518196	0.9933311018341195
1	0.7622999864045775	0.9856512141575198
5	0.7739077326127414	0.9667200289914442
10	0.7756169311200358	0.9592754723948642
15	0.7756169311200358	0.9592754723948642
30	0.7764980147424639	0.9514575649832437
50	0.7764980147424639	0.9514575649832437
70	0.7748413747351482	0.9555792268640245
100	0.7748413747351482	0.9555792268640245

Parametar λ upravlja utjecajem privatnosti u procesu optimizacije - povećanje parametra rezultira povećanjem privatnosti. Međutim, istovremeno dolazi do smanjenja korisnosti (slika 4.4), čime je dokazano postojanje ovisnosti ovih dviju značajki i vrijednost optimizacije. Iz grafičkog prikaza na slici 4.3 vidljivo je kako privatnosti za $\lambda \geq 10$ postaje

gotovo stabilna. Ovo svojstvo potvrđuje kako dodavanje većeg broja novih poveznica, nakon određenog vremena, neće nužno dovesti do povećanja privatnosti.



Slika 4.3 Ovisnost privatnosti o parametru λ



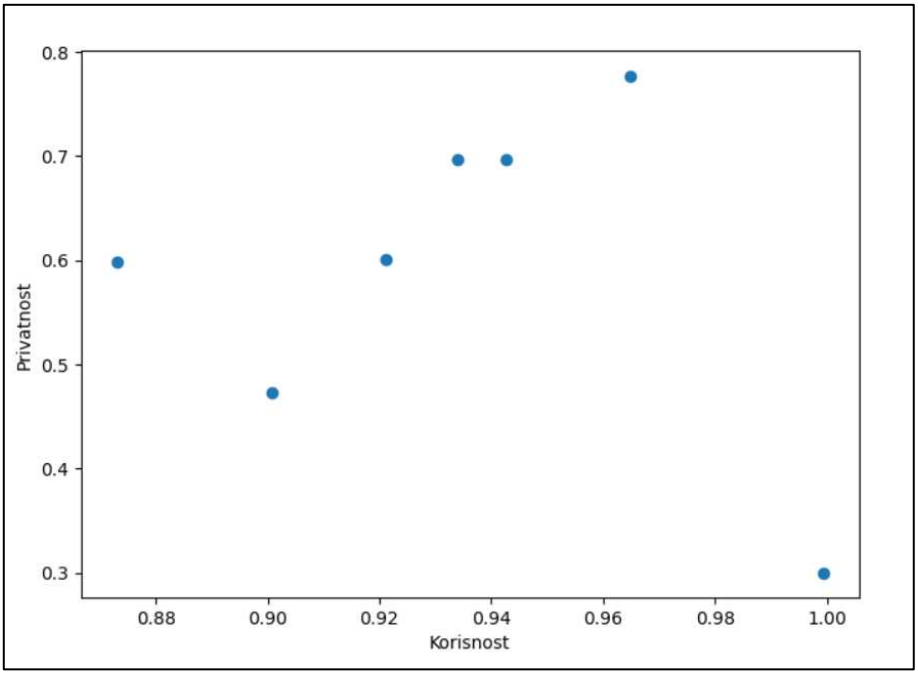
Slika 4.4 Ovisnost korisnosti o parametru λ

Konačno, kako bi odnos privatnosti i korisnosti bio pobliže vrednovan, algoritam pohlepne potrage proveden je nad skupom korisnika s istom veličinom povijesti pretraživanja ($|H^u| =$

25), ali različitom distribucijom poveznica po temama. Korisnici s opisanim karakteristikama nalaze se u tablici 4.1 (*korisnik3 – korisnik9*). Parametar λ postavljen je na vrijednost 10, pošto je pokazano da se tada postiže neka vrsta stabilnosti. Rezultati su prikazani tablicom 4.3 te dodatno slikom 4.5.

Tablica 4.3 Rezultati algoritma za korisnike s $|H^u| = 25$

korisnik	**privatnost**	**korisnost**
korisnik3	0.6974584988600542	0.9425556888355479
korisnik4	0.6974584988600543	0.9425556888355479
korisnik5	0.6003703261228237	0.9212504677584077
korisnik6	0.776967739527151	0.9648877101705449
korisnik7	0.4729032506370176	0.9008474500991136
korisnik8	0.29992115756278714	0.9994211929492189
korisnik9	0.5978921970338859	0.8729763333989555



Slika 4.5 Odnos privatnosti i korisnosti

Iako za sve korisnike vrijedi $|H^u| = 25$, odnos privatnosti i korisnosti za svakog od njih je različit. Do razlike dolazi zbog toga što je povijest pretraživanja nekih korisnika opisana s manje tema. Time je pokazano kako optimizacijski algoritam *PBooster* modela daje bolje rezultate kada je raznolikost interesa korisnika veća, čime povijest pretraživanja sadrži više informacija o korisniku.

4.2. Odabir web-poveznica

Nakon što je poznato koliko poveznica je povoljno dodati u povijest pretraživanja, potrebno je odrediti odakle će se te poveznice preuzeti. U drugoj fazi algoritma *PBooster* provodi se odabir poveznica s ciljem da se dodatno očuva privatnost i kvaliteta personalizirane usluge.

Poznato je kako do povezivanja korisnika na društvenim mrežama dolazi na temelju zajedničkih interesa - korisnik karakteristike dijeli s korisnicima koji se nalaze u njegovoj listi prijatelja. Princip da do socijalnog kontakta u većoj mjeri dolazi između sličnih ljudi naziva se homofilija. Preuzimanje poveznica iz povijesti pretraživanja korisnika koji se nalaze u listi prijatelja rezultiralo bi očuvanjem karakteristika korisnika čija se povijest pretraživanja želi zaštititi, tj. očuvanjem korisnosti. Međutim, takav pristup doveo bi do smanjenja privatnosti i deanonimizacijski napad bio bi učinkovitiji. Prikladnije rješenje je izbor povijesti pretraživanja korisnika koji ne pripadaju listi prijatelja. Beigi et al. [1] predstavljaju rješenje koje će očuvati privatnost, a gubitak korisnosti minimizirati korištenjem matrice odnosa poveznica-tema.

Pretpostavlja se kako korisnik, čija se povijest pretraživanja želi zaštititi, ima barem jedan aktivan profil na nekoj od društvenih mreža. Također, podrazumijeva se da *PBooster* model ima pristup korisnikovoj listi prijatelja na odabranoj društvenoj mreži, $\mathcal{F}^u \neq [\,]$. Prilikom svake promjene ω u vektoru a (promjena koja se događa u fazi odabira tema) odabire se nasumičan korisnik v s javnim profilom na društvenoj mreži koji ne pripada $\mathcal{F}^u$. Za korisnika v simulira se povijest pretraživanja H^v veličine q. Na temelju H^v izgrađuje se pripadna matrica odnosa poveznica-tema T^v te distribucija poveznica po temama c_v. Kako se promjena ω dogodila nad temom t_j, provjerava se postoji li u T^v tema t_j te, ako postoji, je li $c_{vj} \neq 0$. Ako je navedeni uvjet ispunjen, iz matrice T^v se nasumično odabire redak i za koji vrijedi $T^v[i][j] \neq 0$, gdje j predstavlja stupac koji odgovara t_j. Odabrani redak opisuje

poveznicu l koja se zatim dodaje u skup poveznica A. Ako prethodni uvjet nije ispunjen, opisani postupak se ponavlja za novoizabranog nasumičnog korisnika. Kôd 4.1 opisuje fazu odabira poveznica u obliku pseudokôda.

```
Ulaz: F^u, q, a = (a_1, ..., a_m)
Izlaz: skup poveznica A
============================================================
1. A = 0
2. za svaku promjenu ω
3.      neka je t_j odgovarajuća tema promjene ω
4.      izaberi nasumičnog korisnika v, v ∉ F^u
5.      simuliraj povijest pretraživanja H^v veličine q, izvedi T^v i c_v
6.      ako je c_vj = 0, idi na 4. i ponovi, inače
7.          izaberi redak iz T^v[:,][j] vrijednosti različite od 0
8.          izaberi poveznicu l odabranog retka
9.          A = A ∪ l
10.     završi
11. završi
```

Kôd 4.1 Pseudokôd faze odabira poveznica

Konačno, rezultat algoritma je skup poveznica A koji se dodaje u povijest pretraživanja korisnika i jamči optimalnost anonimizacije.

5. Slična rješenja

PBooster model pripada skupu metoda „zagađivanja“. Metode „zagađivanja“ koriste korisnikovu povijest pretraživanja kako bi maksimizirale njegovu privatnost na Internetu. Manipulacija poviješću pretraživanja provodi se dodavanjem skupa poveznica koje generaliziraju karakteristike korisnika, čime ga je teže izdvojiti iz skupine. Uz *PBooster*, iz skupa metoda „zagađivanja“, ističu se anonimizacijski modeli *Random* i *JustFriends*.

5.1. Ostale metode zagađivanja povijesti pretraživanja

Model *Random* manipulira poviješću pretraživanja korisnika tako da dodaje skup nasumičnih poveznica. Prilikom odabira poveznica razmatraju se povijesti pretraživanja korisnika koji ne pripadaju listi prijatelja korisnika čija se privatnost želi zaštititi. Također, za razliku od *PBooster* modela, *Random* provodi odabir poveznica potpuno neovisno o temama koje opisuju promatranu povijest pretraživanja. Ovakav pristup rezultira visokom razinom privatnosti – karakteristike korisnika je vrlo teško raspoznati promatrajući povijest pretraživanja nakon manipulacije. Međutim, mana modela *Random* je što u pitanje dovodi kvalitetu pružane usluge. Iz generalizirane povijesti pretraživanja teško je stvoriti personaliziranu uslugu – dolazi do velikog gubitka korisnosti, tj. zadovoljstvo korisnika rezultatima pretraživanja bit će nisko. Anonimizaciju modelom *Random* prikladno je koristiti u trenucima kada je privatnost korisnika puno važnija od personalizacije rezultata pretraživanja.

Anonimizacija modelom *JustFriends* vrlo je slična modelu *PBooster*. Razlika dvaju modela proizlazi iz činjenice da *JustFriends*, prilikom faze odabira poveznica, koristi povijesti pretraživanja korisnika iz liste prijatelja. Zbog principa homofilije, gubitak korisnosti će biti malen i pružana usluga će ostati prilagođena korisniku. Kako povijest pretraživanja i nakon manipulacije vrlo dobro odražava karakteristike korisnika, deanonimizacijski napadi bit će učinkovitiji, čime je privatnost ugrožena. Ako korisniku veći prioritet predstavlja očuvanje personalizirane usluge, izbor modela *JustFriends* prikladan je za anonimizaciju povijesti pretraživanja.

5.2. Usporedba anonimizacijskih modela

Modeli *Random* i *JustFriends* u središte pozornosti stavljaju jednu od značajki – privatnost ili korisnost, a zanemaruju njihov međusobni utjecaj. *PBooster* uzima ovisnost značajki i predstavlja ju kao važan faktor prilikom anonimizacije povijesti pretraživanja.

Anonimizacija povijesti pretraživanja modelom *Random* pruža vrlo visoku razinu privatnosti i osigurava anonimnost korisnika. Međutim, ako se anonimizacija provodi korištenjem modela *JustFriends*, u povijest pretraživanja bit će očuvani korisnikovi interesi i buduće pretraživanje rezultirat će visokim zadovoljstvom korisnika. *PBooster* model optimizira odnos navedenih značajke – rezultat modela je povijest pretraživanja koja je istodobno dovoljno otporna na deanonimizacijske napade i održava kvalitetu pružane usluge.

Ako je korisnik anonimizirao svoju povijest pretraživanja koristeći *Random,* može biti siguran da se ona ne može povezati s njegovim profilom na društvenim mrežama. *Random* stvara povijest pretraživanja koja je vrlo otporna na napade – dodavanje nasumičnih poveznica dovodi do gotovo jednolike distribucije tema i vrlo je teško izdvojiti korisnika iz skupine. Deanonimizacijski napadi bit će puno učinkovitiji u slučaju kada je korišten *JustFriends* – poveznice ne utječu u velikoj mjeri na distribuciju tema, zbog čega se jedinstvenost korisnika ističe u skupini. Kako je jedinstvenost korisnika ključna u izgradnji personalizirane usluge, model *JustFriends* prikladniji je od modela *Random* kada se spomenuta usluga želi očuvati (ako se teži jednolikoj distribuciji tema u povijesti pretraživanja, tada nije moguće potpuno očuvati personaliziranu uslugu).

U odnosu na *Random*, *PBooster* model stvara povijest pretraživanja koja je manje otporna na deanonimizacijske napade, ali njegova prednost je što nudi očuvanje usluge koja je stvorena na temelju originalne povijesti. Također, *PBooster* je otporniji na napade u odnosu na *JustFriends*, no učinkovitost prilikom očuvanja korisnosti je manja. Kvaliteta *PBooster* modela je što, osim privatnosti korisnika, uzima u obzir i njegovo zadovoljstvo pružanom uslugom. Ovakav pogled na problem zaštite korisnika na Internetu donosi optimalno rješenje koje se ističe među ostalim sličnim metodama vrlo dobrim rezultatima. Optimizacijski problem je NP-težak zbog čega se ne može izvesti učinkovito u konačnom vremenu. Za pronalazak optimalnog rješenja koristi se metoda koja postiže $\frac{1}{3}$ vrijednosti optimalnog rješenja te je rezultat dovoljno dobar i optimizacija se ne smatra velikim nedostatkom. Međutim, složenost raste s porastom količine podataka i pretraživanje može postati neučinkovito.

Zaključak

Potreba korisnika da zaštiti svoju privatnost na Internetu tema je o kojoj se sve više raspravlja. Povijest pretraživanja opisuje interese korisnika, čime postaje glavna meta napada deanonimizacijskih modela. Upotreba raznih alata, kao što su *VPN* i *Tor*, dovodi do željene privatnosti, ali uzrokuje i pad u kvaliteti personalizirane usluge – korisnosti. Anonimizacijski model *PBooster* manipulira poviješću pretraživanja korisnika metodom dodavanja novih poveznica, s ciljem postizanja što veće privatnosti uz očuvanje korisnosti.

U okviru završnog rada predstavljena je uspješna kvantifikacija mjera privatnosti i korisnosti. Privatnost je opisana entropijom dok se gubitak korisnosti definira mjerom sličnosti kosinusa. Optimizacijski problem sveden je na maksimizaciju nemonotone submodularne funkcije kojom je predstavljena kvantifikacija odnosa privatnosti i korisnosti. *PBooster* model problem pronalaska maksimuma opisane funkcije provodi pohlepnim algoritmom lokalne potrage koji rezultira vrijednošću najmanje $\frac{1}{3}$ optimalnog rješenja.

Programski je implementiran pohlepni algoritam koji rezultira optimalnom količinom novih poveznica za svaku temu. Vrednovanje programskog rješenja provedeno je nad skupom umjetno stvorenih podataka. Pokazano je kako povećanje utjecaja privatnosti uistinu dovodi do pada korisnosti. Privatnost postaje gotovo stabilna nakon određenog vremena što ukazuje na to da dodavanje većeg broja novih poveznica ne mora nužno dovesti do njenog povećanja. Također, *PBooster* algoritam je učinkovitiji ako je raznolikost interesa korisnika veća, tj. povijest pretraživanja sadrži više informacija o korisniku.

PBooster model predstavlja vrlo učinkovitu metodu zaštite privatnosti korisnika na Internetu. Prednost nad sličnim metodama očituje se u optimizaciji odnosa značajki privatnosti i korisnosti koju on uvodi. Iako model neće biti jednako učinkovit za sve korisnike, zbog razlike koja proizlazi iz raznolikosti interesa, nedostatak je gotovo zanemariv u usporedbi sa sveukupnom uspješnosti.

Mihaela Svetec

Literatura

[1] Beigi, G., Guo, R., Nou, A., Zhang, Y., Liu, H., *Protecting User Privacy: An Approach for Untraceable Web Browsing History and Unambiguous User Profiles*, *WSDM '19: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, str. 213-221

[2] Su, J., Shukla, A., Goel, S., Narayanan, A., *De-anonymizing Web Browser Data with Social Networks*, *WWW '17: Proceedings of the 26th International Conference on World Wide Web*, (2017), str. 1261-1269

[3] Feige, U., Mirrokni, V.S., Vondrák, J., *Maximizing non-monotone submodular functions*, (2007), poveznica: https://people.csail.mit.edu/mirrokni/focs07.pdf , str. 1-5, pristup: 20.05.2020.

[4] Kulsrestha, R., *A Beginner's Guide to Latent Dirichlet Allocation (LDA)*, (2019), poveznica: https://towardsdatascience.com/latent-dirichlet-allocation-lda-9d1cd064ffa2, pristup: 22.05.2020.

[5] Blei, D.M., Ng, A.I., Jordan, M.I., *Latent Dirichlet Allocation*, *Journal of Machine Learning Research 3*, (2003), str. 993-997

[6] Leskovec, J., Rajaraman, A., Ullman, J.D., Mining of Massive Datasets, (2014), str. 241-262

[7] McCormick, S.T, Iwata, S., *Introduction to Submodular functions*, (2013), poveznica: http://www.iasi.cnr.it/~ventura/Cargese13/Lectures%20slides/Mccormick_Cargese%20intro.pdf, pristup: 10.04.2020.

[8] Delgado-Bonal, A., Approximate Entropy and Sample Entropy: A Compehensive Tutorial, (2019), poveznica: https://www.mdpi.com/1099-4300/21/6/541/htm, pristup: 05.06.2020.

[9] Bansal, S., *Beginners Guide to Topic Modeling in Python*, (2016), poveznica: https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/, pristup: 02.06.2020.

[10] Vodrák, J., predavanje *Polyhedral techniques in combinatorial optimization*, (2010), poveznica: https://theory.stanford.edu/~jvondrak/CS369P/lec16.pdf, pristup: 21.05.2020.

Sažetak

Zaštita korisničke privatnosti metodom anonimizacije povijesti pretraživanja web-preglednika

PBooster model nastao je kao odgovor na potrebu korisnika da zaštiti svoju privatnost prilikom korištenja Interneta. Model provodi anonimizaciju manipulacijom poviješću pretraživanja korisnika tako da u nju dodaje skup novih poveznica. Očuvanje privatnosti i kvaliteta pružane usluge su u obrnuto-proporcionalnom odnosu te se algoritam odabira poveznica temelji na optimizaciji navedenih značajki. U okviru ovog završnog rada predstavljene su glavne značajke i funkcionalnosti *PBooster* modela te deanonimizacijski model koji je bio motivacija za njegovu izgradnju. Programski je implementiran dio algoritma *PBooster* kojim se pronalazi optimalan broj poveznica koje je potrebno dodati u povijest pretraživanja. Provedeno je vrednovanje programskog rješenja nad umjetno stvorenim skupom podataka. Konačno, dana je usporedba PBooster modela sa sličnim metodama anonimizacije.

Ključne riječi: anonimizacija, povijest pretraživanja, privatnost

Summary

Web Browsing History Anonymization Method for User Privacy Protection

The *PBooster* model was created in response to the need for users to protect their privacy when using the Internet. The model performs anonymization with the method of manipulating user browsing history by adding a set of new links to it. Preservation of privacy and quality of provided service are inversely proportional – algorithm for link selection is based on the optimization of these features. Within this final paper, the main features and functionalities of the *PBooster* model, as well as the deanonymization model that was the motivation for its construction, are presented. A part of the *PBooster* algorithm, which finds the optimal number of links that need to be added to the browsing history, has been programmed. An evaluation of the program was performed over the artificially created data set. Finally, a comparison of the *PBooster* model with similar methods of anonymization is given.

Keywords: anonymization, browsing history, privacy

Skraćenice

ISP	*Internet Service Provider*	pružatelj internetskih usluga
VPN	*Virtual Private Network*	virtualna privatna mreža
URL	*Uniform Resource Locator*	ujednačeni lokator sadržaja
LDA	*Latent Dirichlet Allocation*	latentna Dirichletova alokacija