ELEKTROTEHNIŠKI VESTNIK 86(3): 144–151, 2019
IZVIRNI ZNANSTVENI ČLANEK

Analiziranje obnašanja uporabnikov na spletu brez sledenja stanja (z uporabo prstnega odtisa)

Marko Poženel, Matjaž Kukar

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Večna pot 113, 1000 Ljubljana, Slovenija
E-pošta: marko.pozenel@fri.uni-lj.si, matjaz.kukar@fri.uni-lj.si

Povzetek. Razumevanje obnašanja uporabnikov spletnega mesta je ključen del nenehnega izboljševanja spletnega mesta in uporabniške izkušnje. Pri beleženju uporabnikovih akcij lahko trčimo v zakonske omejite, kot sta zakon o varovanju osebnih podatkov in GDPR. Zato je pomembno, da uporabimo pristop, ki je nevsiljiv za uporabnika in omogoča anonimno beleženje akcij uporabnika. Pogosto se kot vir za analizo obnašanja uporabnikov na spletnem mestu uporablja dnevnik spletnega strežnika. Pri tem imamo lahko zaradi slabe strukturiranosti podatkov težave s kakovostjo pridobljenih končnih uporabniških sej. V članku smo predstavili pristop, ki se ukvarja z enim izmed izzivov v fazi preoblikovanja podatkov, izzivom prepletenih sej. To so seje, ki jih določen uporabnik generira s sočasnim brskanjem v več oknih ali zavihkih brskalnika po istem spletnem mestu. Takšne seje kvarno vplivajo na kakovost podatkov in jih je treba analizirati in razplesti. Predstavili smo dva pristopa za razpletanje, ki temeljita na markovskem modelu in preiskovanju prostora stanj. Predstavljena pristopa smo preizkusili na podatkih spletne trgovine. Rezultati kažejo, da lahko na podlagi številke IP in strukture spletne strani verodostojno razpletemo velik del prepletenih sej.

Ključne besede: klikotok, razpletanje sej, hevristično preiskovanje, markovske verige, obnašanje uporabnikov

Analyzing the user browsing behaviour using stateless tracking

Understanding the user' browsing behaviour is a key part of continuous website improvement that enables the delivery of a better user experience. However, identification and tracking a user can violate the legal issues and privacy concerns. It is therefore important to use an approach that is user-unobtrusive and assures anonymous tracking of his/her actions. As a result, the clickstream data obtained from the Web server log files is often used as the main source of data for the user behaviour analysis. However, the data in the Web log files are more or less inadequately structured and therefore eventually leading to an erroneous session reconstruction. In the paper we present an interleaved Web session reconstruction that improves the data quality in the data transformation phase. An interleaved session is a session generated by a particular user concurently browsing the same web site using multiple browser windows or tabs. Such sessions have a detrimental impact on the data quality and have to be analyzed and separated. We propose two approaches for separating interleaved sessions based on the first-order Markov chains and state-space search. The proposed approaches are evaluated on a real-world clickstream data source of a Web shop. The results show that a large amount of the interleaved sessions can be successfully separated on the basis of the IP number and web-site structure.

Keywords: clickstream, session reconstruction, heuristic search, Markov chains, user behaviour

Prejet 19. marec, 2018
Odobren 23. april, 2019

1 UVOD

Svetovni splet je postal eden najpomembnejših virov informacij, tako za zasebne kot tudi za poslovne uporabnike. Omogoča skoraj neomejeno izmenjavo podatkov med različnimi strankami. Podjetja uporabljajo spletne strani za oglaševanje in prodajo svojih izdelkov, institucije za zagotavljanje informacij o svojih storitvah, posamezniki pa za učinkovit dostop do različnih virov informacij. Blogi in socialna omrežja pridobivajo na popularnosti in so že prerasla mejo zasebnega objavljanja. Vedno več ljudi za pridobivanje novic namesto tradicionalnih medijev uporablja družabna omrežja [17]. Obstoječe spletne strani se vsako leto soočajo z več in čedalje boljšimi tekmeci. Vse težje je pritegniti nove stranke in obdržati obstoječe. V takšnih okoliščinah bodo preživele le spletne strani, ki izčrpno razumejo potrebe svojih strank in njihovo obnašanje na spletni strani. Posledično je analiziranje vedenja uporabnikov postalo sestavni del analize spletnih podatkov. Pomembno vlogo ima tako pri vsakodnevnih kot tudi strateških odločitvah. Kakovost odkritih vzorcev obnašanja uporabnikov je zelo odvisna od kakovosti pridobljenih in obdelanih podatkov.

Vedno več tretjih oseb beleži in analizira vedenje uporabnikov na spletu. Po drugi strani pa uporabniki ne želijo sledenja. Nasprotovanje uporabnikov sledenju sta v svojem članku predstavila Mayer in Mitchell [16]. Tehnologije sledenja lahko grobo razdelimo v dve skupini: sledenje s hranjenjem stanja in sledenje brez hranjenja

stanja (angl. fingerprinting). V prvi skupini najdemo navadne piškotke in piškotke HTML5, sledenje geolokaciji in piškotke raznih vtičnikov (npr. Flash). Sledenje brez hranjenja stanja temelji na zaznavi lastnosti brskalnika in ustvarjanju edinstvene oznake na podlagi teh lastnosti. Aktivno sledenje brez hranjenja stanja vključuje aktivno odkrivanje lastnosti oddaljnega odjemalca (vrsta CPE, nastavitve zaslona, pisave ipd). Pasivno sledenje brez hranjenja stanja pa je omejeno na čisti komunikacijski protokol HTML in njegove lastnosti (oddaljeni naslov IP, jezik, napotitelj ipd). Prednost pasivnega sledenja je, da za uporabnika ni preveč vsiljivo, saj uporabnik ne ve, da je predmet proučevanja, niti se mu ne more izogniti.

Eden glavnih virov podatkov za izvedbo analize obnašanja uporabnikov na spletnih straneh so dnevniške datoteke spletnega strežnika (angl. web server log files). Dnevniki spletnih strežnikov so integralni del spletnih strežnikov, vendar prvotno niso bili namenjeni analizi obnašanja uporabnikov. Vsebujejo le omejene, včasih pomanjkljive podatke, njihova vsebina pa ni vedno jasno določena. V članku se ukvarjamo z enim izmed izzivov analize podatkov iz tega vira, prepletenimi uporabniškimi sejami. Predstavili bomo sorodne pristope in predlagali novo tehniko za izboljšanje kakovosti informacij, ki jih pridobimo iz prepletenih sej.

Dnevniške datoteke spletnega strežnika so najpogostejši in najprimernejši vir podatkov o klikotoku (angl. clickstream) [13]. Klikotok je zaporedje dejanj (klikov), ki jih uporabnik naredi med brskanjem po določenem spletnem mestu. Ker je vir podatkov o klikotoku praviloma na voljo, so podatki o klikotoku glavni vir podatkov za analizo obnašanja uporabnikov [27], [2]. Spiliopoulou in sod. [26] so dnevnik spletnega strežnika formalno definirali kot:

Naj množica U označuje vse mogoče spletne zahteve (angl. Web requests) na spletnem mestu. Ta množica vsebuje tako statične strani kot tudi strani, ki jih je mogoče dinamično generirati glede na vhodne parametre. Dnevnik spletnega strežnika L je seznam zahtev strani iz množice U vseh uporabnikov. Zapisi v seznamu L so urejeni po časovni oznaki zahteve.

Dnevniki spletnega strežnika so bili sprva namenjeni za razhroščevanje spletnih strežnikov in ne kot vir podatkov za analize. Zato v dnevnikih spletnih strežnikov pogosto manjkajo nekateri ključni podatki za pridobitev vzorcev obnašanja uporabnikov, posledica tega pa so določene težave z razpoložljivostjo in kakovostjo podatkov [13]. Podatki o klikotoku, pridobljeni iz dnevnikov spletnih strežnikov, so zato lahko nepopolni, vsebujejo šum in ne morejo popolnoma verno zajeti obnašanja uporabnika po spletnem mestu. Zato sta čiščenje in priprava podatkov zahtevna in nagnjena k napakam.

Pri uporabi dnevnikov spletnega strežnika kot vir podatkov lahko naletimo na vrsto težav. Kljub temu zaradi široke dostopnosti in nevsiljivosti ostajajo eden privlačnih virov podatkov. Namenski sistem za prijavo uporabnikov, prilagojen za določeno spletno aplikacijo, je seveda veliko boljša rešitev. Vendar taka rešitev zahteva znatne finančne, računalniške in človeške vire. Poleg tega je treba načrtovati dovolj vnaprej, da pravočasno zberemo želene količine podatkov o obnašanju uporabnikov, tako na strežniški strani kot na strani odjemalca. Takšen način zbiranja podatkov lahko povzroča tudi zakonske težave zaradi zasebnosti podatkov (npr. pomisleki glede uporabe piškotkov [12], [25]). Z uporabo izključno dnevnika spletnega strežnika se izognemo tem težavam, saj je beleženje dostopov sestavni mehanizem spletnega strežnika, relativno nezahteven za vire, na voljo pa je ogromna količina že zbranih zgodovinskih podatkov. Zato večina študij analize spletnih uporabnikov uporablja kot vir za podatke dnevnik spletnega strežnika [3], [29].

2 PREGLED PODROČJA

Uporabniško sejo sestavlja zaporedje akcij, ki jih uporabnik izvede med brskanjem po spletnem mestu v določenem časovnem obdobju. Podatki o sejah se največkrat v razpršeni obliki beležijo v dnevniške datoteke. Pri rekonstrukciji sej podatke pretvorimo v obliko, ki je primerna za nadaljnjo uporabo [5], [21].

Verna rekonstrukcija uporabniških sej ni lahka naloga, ker HTTP-protokol nima stanja in individualne zahteve za spletne vire niso povezane med sabo (tj. zahteve nimajo identifikatorja seje). Pri reaktivnem pristopu rekonstrukcije sej [6], [7] identificiramo uporabnike na podlagi dnevnikov brez uporabe drugih podatkov z odjemalca (npr. piškotki). V takem primeru bodo uporabniki, ki so skriti za strežnikom proxy ali pa sočasno uporabljajo več zavihkov spletnega brskalnika, v dnevniških datotekah kreirali seje, ki imajo enak IP in bodo zato lahko obravnavani kot en sam uporabnik.

Pri uporabi reaktivnih strategij za rekonstrukcijo sej so soočamo s pomanjkanjem nekaterih podatkov (npr. identifikator seje), zato se poslužujemo različnih hevrističnih pristopov. Izzivi, na katere naletimo pri uporabi te metode, so:

- Identifikacija posameznih uporabnikov. Identifikacija uporabnika glede na naslov IP je lahko netočna. Izboljšamo jo lahko z vključitvijo podpisa brskalnika (atribut user-agent) v dnevnik [6]. Mehanizmi, ki posegajo v zasebnost uporabnika, zaradi zaščite zasebnosti niso zaželeni [16].
- Identifikacija spletnih sej iskalnih robotov, še zlasti, če spletni dnevnik ne hrani podpisa brskalnika. Identificiramo jih lahko na podlagi njihovega podpisa v atributu *u*ser-agent, preverbo naslova IP v podatkovni zbirki spletnih robotov ali drugih značilnostih [18].
- Verna rekonstrukcija aktivnosti posameznih uporabnikov, kjer so posamezne zahteve dodeljene pravim uporabnikom. V literaturi so bile predlagane različne hevristike: časovno usmerjena hevristika [6], navigacijsko usmerjena hevristika [7], pristop

z uporabo linearne optimizacije (angl. Integer programming) [8], uporaba navigacijskih zaporedij [1].
- Obravnavanje uporabe gumba za nazaj v brskalniku. Akcija *n*azaj se ne zabeleži nujno v dnevniški datoteki spletnega strežnika [9], [18]. Manjkajoči zapisi v dnevniški datoteki pa lahko negativno vplivajo na rekonstrukcijo sej in lahko povzročijo nepravilno rekonstrukcijo sej.
- Vzporedno brskanje v več zavihkih brskalnika [11], [19]. Postopek rekonstrukcije sej mora upoštevati scenarij, v katerem ima uporabnik odprtih več zavihkov ali oken brskalnika za brskanje po istem spletnem mestu. Za uporabnika vsak zavihek običajno pomeni ločeno uporabniško sejo. Takšno obnašanje je pogosto za napredne uporabnike in povzroča *prepletene seje*. Ker se spletni strežnik ne zaveda različnih zavihkov, se take seje na strežniški strani vidijo kot ena dolga seja in jih je treba ustrezno razplesti [21].

Veliko zgornjih izzivov lahko rešimo z uporabo proaktivnih strategij [10], [24], kot je uporaba piškotkov, ali pa z dinamičnim generiranjem zahteve za spletno stran na odjemalcu, ki ji pripnemo identifikator uporabnika. Pomanjkljivost tega pristopa je, da moramo spremeniti strukturo spletne strani, kar pomeni poseganje v obstoječo rešitev. Ena od aktivnih strategij je tudi uporaba sistemov za sledenje tretjim osebam (orodje Web Analytics, npr. Google Analytics), kjer se v vsako spletno stran vstavi JavaScript koda tretjih oseb. Slabost tega pristopa so varnostna vprašanja, saj se podatki o uporabi spletnih strani običajno pošiljajo na spletno mesto tretje osebe, kjer se obdelujejo in analizirajo [1]. Uporaba storitev tretjih oseb poleg vrednosti prinaša tudi pomisleke glede zasebnosti. Spletni uporabniki so namreč čedalje manj naklonjeni uporabi piškotkov, posredovanju podatkov tretjim osebam in uporabi njihovih vzorcev brskanja za namene oglaševanja [16]. Zato se lastniki spletnih mest raje izogibajo uporabi proaktivnih strategij.

Nobeden izmed predstavljenih pristopov ne rešuje problema prepletenih sej. Kolikor vemo, je edini pristop, ki se ukvarja s problemom vzporednega brskanja in prepletenih sej, predstavil Viermetz s sod. [28]. Vendar je njihov pristop usmerjen k boljšemu razumevanju vedenja uporabnikov spletnega mesta, manj poudarka je na samem postopku razpletanja. Želeli smo zapolniti vrzel in razviti izviren pristop za razpletanje prepletenih sej, ki temelji na markovskih verigah prvega reda in omogoča učinkovito razpletanje prepletenih sej.

3 PREDSTAVITEV PODATKOV

Predobdelani podatki zajemajo podatke o klikotoku, združenih v uporabniške seje, in načrt spletnega mesta.

Načrt spletnega mesta je graf spletnih strani iz množice *U*, ki so dostopni uporabnikom in med sabo povezani s hiperpopezavami. Neposredna povezava med spletnima stranema x_i in x_j v načrtu pomeni večjo verjetnost prehoda uporabnika med tema dvema stranema, kot če neposredne povezave ni. Načrt spletnega mesta lahko zelo olajša analizo podatkov, zlasti kadar so spletne strani med sabo zelo redko povezane. Uporabniško sejo S lahko predstavimo kot zaporedje spletnih strani

$$S = [x_1, x_2, \ldots, x_k], \tag{1}$$

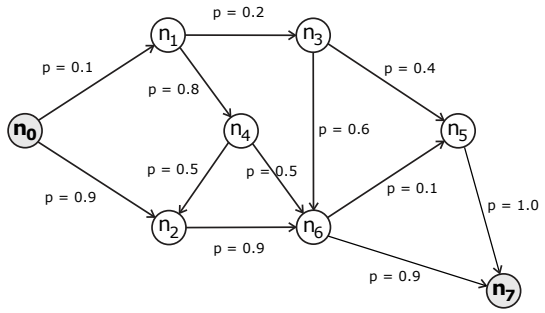
kjer stran x_1 pomeni začetno (vstopno) spletno stran in x_k zadnjo obiskano spletno stran v uporabniški seji. Posamezno uporabniško sejo lahko obravnavamo kot pot skozi graf, ki v našem primeru predstavlja načrt spletnega mesta. Pot se začne v začetnem vozlišču, ki ustreza prvi strani x_1 v seji, in konča v zadnjem vozlišču, ustreza zadnji obiskani strani x_k uporabniške seje.

Postopek razpletanja sej temelji na vzorcih, pridobljenih iz preteklih uporabniških sej. Surove podatke o klikotoku iz dnevnika spletnega strežnika je treba preoblikovati v obliko, primerno za nadaljnjo obdelavo. Vsaka uporabniška seja v klikotoku je predstavljena kot zaporedje zahtevanih strani. Klikotok lahko preoblikujemo v graf, kjer vsaka uporabniška seja predstavlja pot skozi graf. Vozlišča v grafu pomenijo zahtevane spletne strani, povezave pa prehode med temi stranmi.

Začnemo s praznim grafom, ki vsebuje vsa vozlišča iz množice *U*, in postopoma dodajamo povezave glede na prehode v uporabniških sejah. Če uporabljamo predznanje (npr. načrt spletnega mesta), začnemo z grafom, ki vsebuje povezave v skladu s predznanjem (npr. povezave, ki se nahajajo v načrtu spletnega mesta). Na podlagi dveh sosednjih strani x_i, x_{i+1} v zaporedju uporabniške seje povežemo ustrezni dve vozlišči n_i in n_{i+1} v grafu in nastavimo utež povezave $w(n_i, n_{i+1})$. Če sta vozlišči n_i in n_{i+1} že povezani, samo ustrezno posodobimo utež povezave $w(n_i, n_{i+1})$. Več ko je poti v grafu, ki si deli odsek poti $n_i \rightarrow n_{i+1}$, večja je utež $w(n_i, n_{i+1})$. Ko graf posodobimo s podatki vseh uporabniških sej, dobimo utežen graf, ki ustreza klikotoku. Utežene poti grafa lahko uporabimo za izračun verjetnosti prehoda med sosednjimi vozlišči. Vsako vozlišče n_i je povezano z več vozlišči n_j in za vsako povezavo poznamo verjetnost prehoda v vozlišče n_j, tj. $P(n_i \rightarrow n_j)$.

Markovski modeli se pogosto uporabljajo za modeliranje časovno odvisnih zaporednih problemov. Lahko jih uporabimo za napovedovanje najverjetnejšega naslednjega stanja, izračun verjetnosti opazovanega zaporedja ali za iskanje najverjetnejšega zaporedja.

Markovska veriga je definirana kot zaporedje naključnih spremenljivk, s_0, s_1, s_2, …, ki imajo markovsko lastnost. To pomeni, da je prihodnost pogojno neodvisna od preteklosti. Markovsko verigo lahko uporabimo pri izračunu verjetnosti določene uporabniške seje. Vsako stanje (označeno z s_i) v markovski verigi ustreza spletni strani (označeno z x_i). Dogodki (kliki hiperpovezav) ustrezajo prehodu iz enega stanja v drugo. Pri razpletanju smo uporabili markovsko verigo prvega reda. Verjetnosti prehodov med stanji smo določili iz



Slika 1: Markovska veriga spletnega mesta. Vozlišče n_0 je začetna stran, n_7 pa končna stran. Prehodi med stanji označujejo verjetnosti prehodov med stranmi. Verjetnost poti $n_0 \to n_2 \to n_6 \to n_7$ je $0.9 * 0.9 * 0.9 = 0.729$.

preteklih uporabniških sej, kjer smo markovsko verigo naučili s podatki čistih, nepreplетenih sej. Slika 1 prikazuje markovsko verigo spletnega mesta.

4 METODA RAZPLETANJA

V naših zgodnejših poskusih smo razvili osnovno metodo za razpletanje sej, ki uporablja markovski model prvega reda [20]. Temelji na požrešnem pristopu, ki zaporedno razporeja strani prepletene seje v delno razpletene seje. Temelji na dejstvu, da prehod med stranema prepletene seje $x_i \to x_{i+1}$ z večjo verjetnostjo pripada eni izmed razpletenih sej. Pomanjkljivost te metode je, da ne zagotavlja najbolj verjetnih razpletenih sej. Za zagotovitev optimalnejše rešitve med drugim lahko uporabimo metode preiskovanja v prostoru stanj. Našo osnovno metodo lahko premočrtno obravnavamo kot požrešno iskanje lokalno optimalne rešitve v prostoru stanj.

Obstaja veliko pristopov za iskanje rešitev v prostoru stanj. Osnovni iskalni strategiji sta iskanje v *globino* in iskanje v *širino*. Iskanje v globino ima linearno pomnilniško zahtevnost glede na dolžino poti, vendar ne najde nujno rešitve. Iskanje v širino preišče celoten prostor stanj, zato vedno najde vse rešitve in tudi optimalno. Pomanjkljivost iskanja v širino je velika prostorska zahtevnost, ker mora v pomnilniku hraniti vsa vozlišča, da lahko generira vozlišča na naslednji ravni. Če je na voljo znanje o problemski domeni, ki lahko usmerja iskanje, lahko uporabimo metode informiranega iskanja, kot je na primer best-first search [4], ki iskanje usmerja v smeri najbolj perspektivnega vozlišča. V tem članku bomo predstavili metodo za razpletanje sej, ki uporablja informirano iskanje v širino (RBFS).

4.1 Prostor stanj za razpletanje sej

Prostor stanj lahko predstavimo kot usmerjen graf, katerega vozlišča (stanja) ustrezajo rešitvam delnih problemov, povezave med vozlišči pa ustrezajo legalnim prehodom med stanji. Problem iskanja se pretvori v problem iskanja poti med začetnim stanjem (začetno vozlišče) in končnim stanjem (končno vozlišče). Prehodom med stanji lahko pripišemo ceno. V našem primeru nas zanimajo rešitve z minimalno ceno. Prostor stanj za razpletanje sej je graf oz. bolj natančno drevo. Začnemo z dobro definiranim začetnim stanjem, iz katerega potem rekurzivno izpeljemo delne razplete po nivojih drevesa, ki tvori prostor stanj.

Problem razpletanja sej lahko formuliramo kot problem iskanja v prostoru stanj. Vsako stanje Z pomeni prepleteno sejo

$$S_I = [x_1, x_2, \ldots, x_n]$$

z dolžino n z r delno razpletenimi sejami S_S,

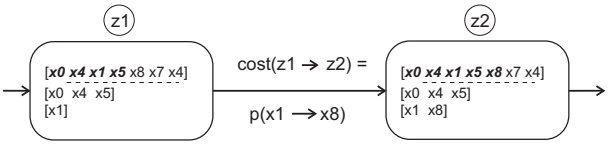
$$\begin{aligned} Z &= \langle [S_S^{(1)}, S_S^{(2)}, \ldots, S_S^{(r)}],\ S_I \rangle \\ &= \langle [(x_1^{S_S^{(1)}}, x_2^{S_S^{(1)}}, \ldots, x_a^{S_S^{(1)}}), \\ &\qquad \vdots \\ &\qquad (x_1^{S_S^{(r)}}, x_2^{S_S^{(r)}}, \ldots, x_b^{S_S^{(r)}})], \\ &\quad (x_j, x_{j+1}, \ldots, x_n) \rangle, \end{aligned}$$

kjer $S_S^{(i)}$ pomeni i-to delno razpleteno sejo, S_I prepleteno sejo (razpleteno do elementa x_i) in x_k stran v seji. Začetno stanje je predstavljeno kot

$$Z_S = \langle [], (x_1, x_2, \ldots, x_n) \rangle$$

s praznim seznamom na začetku, kar pomeni, da nimamo še nobenih delno razpletenih sej, in zaporedjem strani, ki pomeni celo prepleteno sejo. Če imamo prepleteno sejo z dolžino n, lahko razpletanje vrne največ n razpletenih sej. Če domnevno prepletena seja sploh ni prepletena, dobimo eno samo razpleteno sejo.

Pri razpletanju se premikamo med stanji po povezavah, ki imajo pozitivne uteži (cena). Prehod iz stanja v sosednje stanje ustreza bodisi dodeljevanju naslednje strani prepletene seje eni od delno razpletenih sej ali začetku nove razpletene seje. Akcije dodeljevanja strani ne moremo razveljaviti, posledica pa je velik prostor stanj v obliki drevesa. Število naslednikov vsakega vozlišča je odvisno od števila začetih razpletenih sej v trenutnem vozlišču. Vsak prehod dodeli naslednjo stran prepletene seje x_i na konec ene izmed delno razpletenih sej S_S ali pa začne novo sejo $S_S^{(r+1)}$. Slika 2 prikazuje primer dveh stanj Z_1 in Z_2 in akcije, ki označuje prehod med stanjema.



Slika 2: Primer prehoda med dvema stanjema v našem prostoru stanj. Zaporedje nad črtkano črto pomeni prepleteno sejo. Krepko napisane strani so že bile dodeljene delno razpletenim sejam pod črtkano črto. Akcija doda naslednjo stran prepletene seje x_8 na konec druge razpletene seje.

Primer prehoda med dvema državama v našem iskalnem prostoru. Zaporedje nad črtkano črto pomeni prepleteno sejo s krepkimi stranmi, ki so že dodeljene delno ločenim sejam pod pikčasto črto. Dejanje doda naslednjo stran od prepletene seje (x_8) do konca druge ločene seje.

4.2 Hevristično preiskovanje

Če želimo razplet prepletene seje, da je produkt verjetnosti prehodov med stranmi največji, moramo načeloma preiskati celoten prostor stanj. Skupno število vozlišč drevesa stanj z n nivoji je enako $\sum_{k=0}^{n} B_k$ in raste eksponentno v odvisnosti od v n. Preiskovanje vseh poti v drevesu stanj je časovno izjemno potratno zaradi eksponentnega večanja alternativ na vsakem nivoju drevesa, kar kliče po uporabi problemsko specifične hevristike. Namen hevurističnega iskanja je s pomočjo funkcije $f(Z) = g(Z) + h(Z)$ ovrednotiti, katera vozlišča množice kandidatov so najbolj obetavna, in naprej iskati v smeri najbolj obetavnega. Pri tem $h(Z)$ pomeni hevristično ocena obetavnosti vozlišča in $g(Z)$ ceno najkrajše poti od začetnega vozlišča do vozlišča Z. Zaradi linearne prostorske zahtevnosti smo uporabili prostorsko učinkovito implementacijo algoritma A^* – algoritem recursive best-first search (RBFS) [14].

4.2.1 Dopustnost hevristične funkcije: je zaželena lastnost hevristične ocenjevalne funkcije. Naj za vsako vozlišče Z v prostoru stanj velja, da je $h^*(Z)$ cena optimalne poti od vozlišča Z do ciljnega vozlišča. Izrek o popolnosti trdi, da je hevristična funkcija dopustna (optimistična), če za vsa vozlišča Z v prostoru stanj velja

$$h(Z) \leq h^*(Z).$$

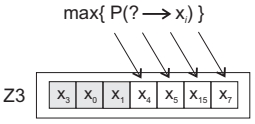
Ko pri iskalnih algoritmih iz družine A^* (vključno RBFS) za usmerjanje iskanja uporabimo dopustno hevristično funkcijo, iskalni algoritmi vedno najdejo optimalno rešitev (najcenejšo pot) [22]. Nedopustne (pesimistične) hevristične funkcije ne zagotavljajo iskanja optimalnih rešitev, lahko pa delujejo hitreje [4].

4.2.2 Hevristična funkcija za razpletanje: Za problem razpletanja prepletenih sej nas zanima ciljno vozlišče, ki maksimira produkt verjetnosti prehodov med stranmi v vseh razpletenih sejah. Dopustna hevristična funkcija $h(Z)$ mora optimistično oceniti razplet preostalega dela prepletene seje. Maksimirati mora produkt verjetnosti prehodov med stranmi (ali enakovredno, zmanjšati vsoto njihovih negativnih logaritmov), ki še niso bile razporejene delno razpletenim sejam.

Trivialna dopustna hevristična funkcija za problem razpletanja sej, je kar

$$h_0(Z) = 1 * 1 * \ldots * 1 = 1 \qquad (2)$$

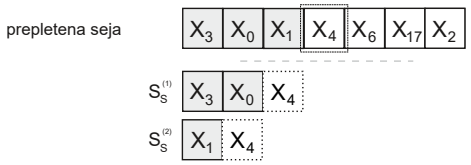
Taka hevristična funkcija bi obravnavala vsa vozlišča enako, vendar ne bi pripomogla k smiselnemu usmerjanju iskanja. Za dobro vodenje mora biti $h(Z)$ čim bliže $h^*(Z)$. Dopustna hevristična funkcija h, ki ponuja dobro usmerjanje, je predstavljena v nadaljevanju.



Slika 3: Vozlišče $Z3$ hrani stanje delno razpletene prepletene seje.

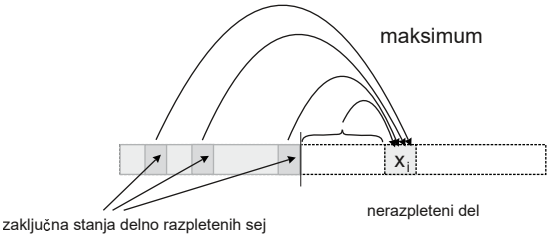
Oglejmo si sliko 3, ki prikazuje vozlišče $Z3$. Prepletena seja je delno razdeljena na dve začeti razpleteni seji, kot prikazuje slika 4. Prvo stran prepletenega dela (stran x_4) lahko bodisi dodamo na konec ene izmed delno razpletenih sej bodisi začnemo novo razpleteno sejo. Ko računamo vrednost $\max\{P(? \rightarrow x_i)\}$, ne smemo upoštevati vseh strani v razporejenem delu, ampak samo zadnjo stran vsakega delno razpletenega dela. Pri vozlišču $Z3$ in dodelitve strani x_4 je izračun enak:

$$\max\{P(? \rightarrow x_4)\} = \max\{P(x_0 \rightarrow x_4), P(x_1 \rightarrow x_4)\}$$



Slika 4: Delno razpletena prepletena seja. Naslednjo nerazporejeno stran prepletene seje (x_4) lahko dodamo na konec enega od začetih delnih razpletov.

Pri izračunu vrednosti hevristične funkcije moramo upoštevati vse strani, ki se v še nerazpletenem delu prepletene seje nahajajo pred stranjo x_i. Postopek za izračun izraza $\max\{P(? \rightarrow x_i)\}$ je prikazan na sliki 5. Hevristika h je dopustna pod pogojem, da lahko nedvoumno določimo začetne strani razpletenih sej. Če ta pogoj ni izpolnjen, bo rezultat morda vseboval preveč razpletenih sej. Pokazalo se je, da je predstavljena hevristična funkcija h izjemno učinkovita pri usmerjanju iskanja za problem razpletanja prepletenih sej, zato smo jo uporabili tako pri sintetičnih kot tudi pri realnih podatkih.



Slika 5: Izračun izraza $\max\{P(\rightarrow x_i)\}$ hevristične funkcije

Tabela 1: Rezultati razpletanja sej na podlagi podatkov spletne trgovine

Metoda vrednotenja	Požrešno	RBFS
Popolno ujemanje	0.334	0.398
WLCS	0.504	0.541

Tabela 2: Delež kupcev v prepletenih in neprepletenih sejah.

seje	št. sej	št. kupcev	% kupcev
neprepletene	10353	1751	17
prepletene	9343	2792	30

5 REZULTATI

Umetno generirana množica prepletenih sej je sestavljena iz različnega števila preverjenih, čistih sej. Tretjina prepletenih sej vsebuje dve čisti seji, ena tretjina tri čiste seje, preostala tretjina pa so neprepletene čiste seje. Tako generirane prepletene seje odražajo realno obnašanje uporabnikov in omogočajo testiranje zmogljivosti postopka razpletanja v odvisnosti od števila vsebovanih sej. Po razpletanju smo primerjali podobnost dobljenih sej z originalnimi sestavnimi sejami. Za ocenjevanje podobnosti smo uporabili kriterija popolnega ujemanja in najdaljšega skupnega podzaporedja (WLCS) [15]. Popolno ujemanje je zelo natančna metoda, WLCS pa se izkaže kot zelo uravnotežena metoda.

Rezultate ovrednotenja razpleta sej spletne trgovine prikazuje tabela 1. Vrstice označujejo metodo vrednotenja (popolno ujemanje, WLCS), stolpci tabele pa metode razpletanja. Vrednosti v tabeli pomenijo povprečno podobnost med razpletenimi in dejanskimi sejami in se gibljejo med 0 (popolnoma različni) in 1 (popolno ujemanje).

Prva vrstica tabele vsebuje povprečno podobnost po metodi popolnega ujemanja. Seje so bodisi popolnoma pravilno razpletene (podobnost zaporedij strani v referenčni in razpleteni seji je 1) bodisi napačno razpletene (podobnost enaka 0). Pri vrednotenju razpletanja po tej metodi že ena sama napačno razporejena stran v uporabniški razpleteni spletni seji pomeni napačen razplet. Povprečna podobnost za popolno ujemanje torej je odstotek vseh razpletenih sej iz množice prepletenih sej, ki so bile popolnoma pravilno razpletene. Rezultat 0.334 torej pomeni, da je bilo 33.4% razpletenih sej po požrešni metodi (Greedy) razpletenih popolnoma pravilno. Podobno velja za metodo preiskovanja prostora stanj s hevristično funkcijo, kjer je bilo popolnoma pravilno razpletenih 39.8% sej. Druga vrstica tabele vsebuje rezultate vrednotenja razpletanja po metodi najdaljšega skupnega podzaporedja (angl. WLCS). Rezultat vrednotenja predstavlja podobnost sej na podlagi najdaljšega skupnega podzaporedja strani v obeh sejah. Torej, če dve seji nimata skupnih strani, je rezultat 0. Če imata dve seji polovico strani zaporedja v istem vrstnem redu, je rezultat 0.5, če pa so vse strani obeh sej v istem vrstnem redu, je podobnost enaka 1.

Razpletanje prepletenih sej spletne trgovine je izziv zaradi velikega števila spletnih strani in neenotne vstopne točke. Načrt strani spletnega mesta trgovine je velik, kar pomeni, da obstaja veliko (potencialnih) uporabniških poti po spletnem mestu. Z združevanjem strani v skupine smo zmanjšali število vhodov v markovski model na obvladljivo raven, izgubili pa smo del informacije o individualnih sejah. Poleg tega je vstopna (začetna) stran uporabnika lahko skoraj poljubna stran spletnega mesta (npr. povezave s pasic), kar prinaša težjo identifikacijo začetnih strani v prepletenih sejah. To pomeni, da je kršena predpostavka o popolnosti (angl. admissibility), da lahko zanesljivo zaznamo začetne strani. Zato hevristično preiskovanje prostora stanj (RBFS) pogosto daje neoptimalne rešitve.

Slika 6 prikazuje podrobne rezultate vrednotenja za obe metodi razpletanja in obe metodi vrednotenja. Vsak graf prikazuje rezultate za eno metodo razpletanja in eno metodo vrednotenja razpletanja. Stolpci predstavljajo metode vrednotenja (tj. požrešna in preiskovanje prostora stanj), vrstice pa metode razpletanja (tj. popolno ujemanje in WLCS). Vsak stolpični diagram je histogram, ki prikazuje porazdelitev rezultatov razpletanj sej. Na osi x so navedeni intervali podobnosti sej, na osi y pa sta navedena število in odstotek razpletenih sej po posameznih intervalih podobnosti. Črtkana črta prikazuje končni rezultat (povprečje podobnosti vseh sej), ki je naveden v tabeli 1.

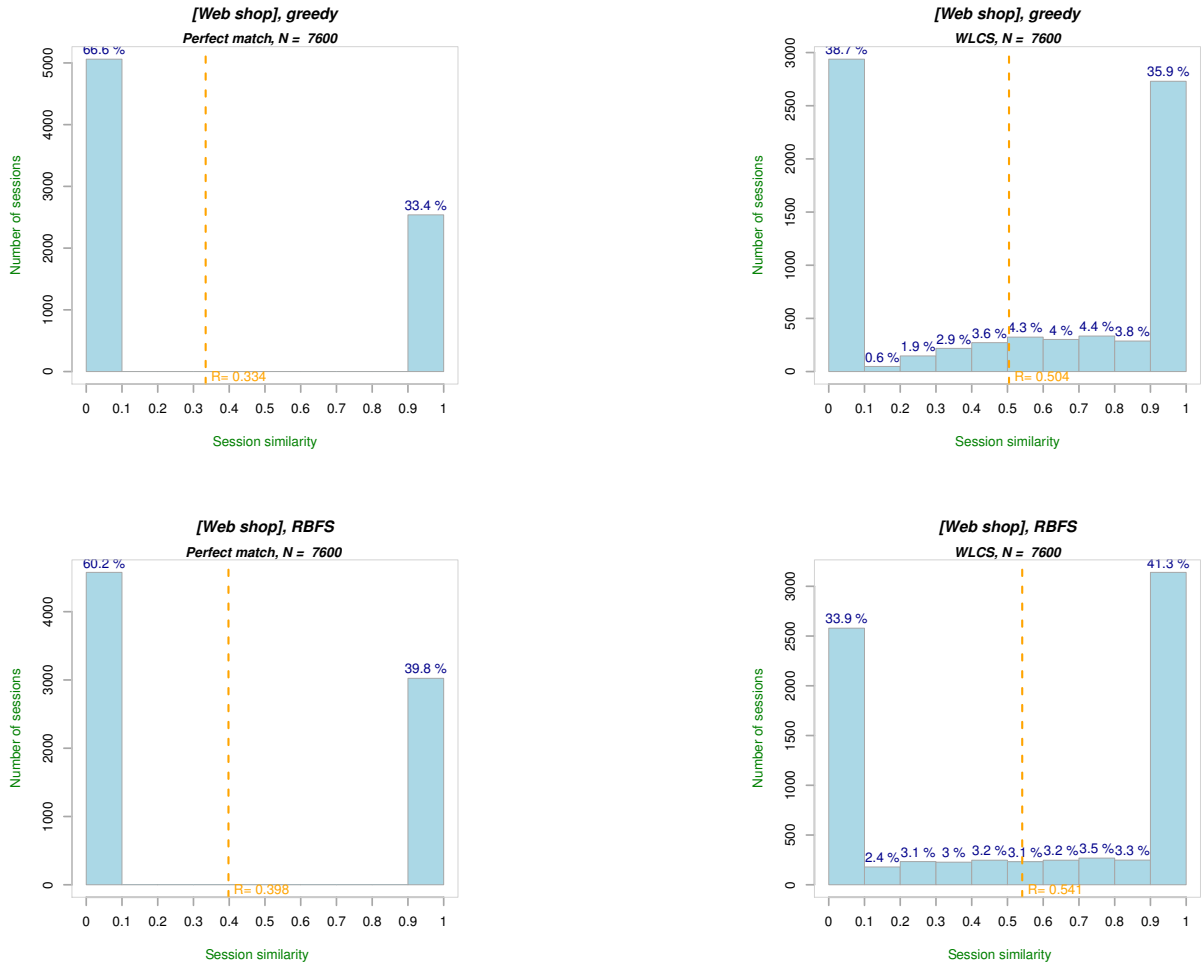
Grafi na sliki 6 poleg končnega rezultata prikazujejo razporeditev podobnosti med referenčnimi in razpletenimi sejami. Porazdelitev je lahko enako pomembna kot sam končni povprečni rezultat, saj končni rezultat ne pokaže celotne slike. Veliko število razpletenih sej s podobnostjo 0.9 je veliko boljše kot veliko število sej s podobnostjo 0.1. Idealen graf (histogram) rezultata razpletanja bi imel en sam stolpec s podobnostjo vseh razpletenih sej med 0.9 in 1.0.

Rezultate razpletanja smo testirali s t-testom za odvisne vzorce (parni t-test). Test je pokazal, da je razpletanje s preiskovanjem prostora stanj (RBFS) statistično značilno boljša metoda razpletanja kot požrešna metoda ($p < 0.01$).

Zanimale so nas tudi značilnosti uporabnikov, ki generirajo prepletene seje. Analizirali smo uporabniške seje, ki smo jih lahko nedoumno identificirali na podlagi uporabnikove prijave. Rezultate analize prikazuje tabela 2, kjer vrstice označujejo vrsto sej. Delež prepletenih sej, ki ga generirajo kupci, je skoraj dvakrat večji od tistih, ki ga generirajo drugi uporabniki.

6 DISKUSIJA IN SKLEP

Danes sta na spletu praviloma povsod prisotna sledenje uporabnikom in hranjenje podatkov o njihovih akcijah za



Slika 6: *Rezultati razpletanja sej spletne trgovine. Uporabili smo dve metodi za merjenje podobnosti med dvema sejama: popolno ujemanje in WLCS.*

izboljšanje uporabniške izkušnje ali prikazovanje ustrezne vsebine (npr. oglasov). Za uporabnike je to lahko moteče in sproža čedalje več vprašanj s področja zaščite zasebnosti. Zakonodajalci si prizadevajo ustvariti pravne okvire, ki strogo omejujejo tehnike sledenja. Zato je za razumevanje obnašanja uporabnikov smiselno uporabiti pristope, kot je predlagan v tem članku, ki niso moteči za uporabnika in ne posegajo v njegovo zasebnost.

V tem delu smo predstavili metodo za izboljšanje postopka obdelave podatkov o klikotoku. Metoda omogoča razpletanje prepletenih spletnih sej. Vključuje uporabo markovske verige prvega reda, načrta spletne strani (angl. web site map) in iskalno strategijo prostora stanj na podlagi hevristike. Predstavili smo motivacijo, delovanje metode in ovrednotili metodo v praksi na realnem viru podatkov spletne trgovine. Predlagana hevristika za usmerjanje iskanja se je izkazala za uspešno, saj dosega uporabne rezultate. Metoda samodejno določi najverjetnejše število prepletenih sej, zato jo lahko uporabimo ne glede na število sestavnih sej v prepletu. Uporabimo jo lahko za različnie vire podatkov o klikotoku, vendar lahko pričakujemo boljše rezultate za dobro definirane uporabniške seje z manj vstopnimi točkami v sistem. Predstavljena hevristična metoda iskanja deluje po načelu iskanja najverjetnejšega zaporedja strani, kar pa ni vedno ustrezno (seje z manj verjetnimi prehodi med stranmi).

Predlagan pristop je koristen za poglobitev ozaveščenosti o obstoju prepletenih spletnih sej, ki so značilne za napredne spletne uporabnike. Z razpletanjem takšnih sej izboljšamo kakovost podatkov o klikotoku za to pomembno skupino uporabnikov. Rezultati kažejo, da na podlagi strukture spletne strani in številke IP kot identifikatorja uporabnika verno razpletemo velik del prepletenih sej.

Predpostavka našega pristopa je, da pričakujemo pravilno identifikacijo začetnih strani. Če začetne strani ni mogoče dovolj dobro določiti, hevristična funkcija teži k pesimističnemu ocenjevanju. V posebnih (večinoma realnočasovnih) primerih lahko pesimistične hevristične

funkcije premagajo optimistične (popolne) [22], [23]. V velikih prostorih stanj (kot je naš) je pogosto bolje imeti dobro pesimistično hevristično funkcijo kot skoraj neinformativno optimistično. Tak primer je tudi spletna trgovina, kjer so začetne strani samo verjetnostno določene, kljub temu pa še vedno dobimo koristne rezultate.

Najbolj spodbuden rezultat je nedvomno ugotovitev, da bodo uporabniki, ki generirajo prepletene seje, z dvakrat večjo verjetnostjo kaj kupili v primerjavi s tistimi, ki takih sej ne generirajo. To pomeni, da je lahko že zaznavanje vzporednega brskanja uporabnikov uporabno in podlaga za nadaljnjo prilagoditev vsebine spletnih strani.

LITERATURA

[1] Murat Ali Bayir, Ismail Hakki Toroslu, Murat Demirbas and Ahmet Cosar. Discovering better navigation sequences for the session construction problem. *Data & Knowledge Engineering*, 73:58–72, 2012.

[2] Fabrício Benevenuto, Tiago Rodrigues, Meeyoung Cha and Virgílio Almeida. Characterizing user navigation and interactions in online social networks. *Information Sciences*, 195:1–24, 2012.

[3] B. Berendt, B. Mobasher, M. Nakagawa and M. Spiliopoulou. The impact of site structure and user environment on session reconstruction in Web usage analysis. In *WEBKDD - KDD Workshop on Web Mining and Web Usage Analysis*, pages 159–179, 2002.

[4] I. Bratko. *Prolog Programming for Artificial Intelligence*. Pearson Addison-Wesley, Harlow, England, 3. edition, 2000.

[5] V Chitraa, Dr. Davamani and Antony Selvdoss. A survey on preprocessing methods for web usage data. *arXiv preprint arXiv:1004.1257*, 2010.

[6] R. Cooley, B. Mobasher and J. Srivastava. Data Preparation for Mining World Wide Web Browsing Patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.

[7] Robert Cooley, Pang-Ning Tan and Jaideep Srivastava. Discovery of interesting usage patterns from web data. In *Web Usage Analysis and User Profiling*, pages 163–182. Springer, 2000.

[8] Robert F Dell, Pablo E Román and Juan D Velásquez. Web user session reconstruction using integer programming. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 385–388. IEEE Computer Society, 2008.

[9] Robert F. Dell, Pablo E. Román, and Juan D. Velásquez. Web user session reconstruction with back button browsing. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pages 326–332. Springer, 2009.

[10] Yongjian Fu and Ming-Yi Shih. A framework for personal web usage mining. In *International Conference on Internet Computing*, pages 595–600, 2002.

[11] J. Huang and R. W. White. Parallel browsing behavior on the Web. In *Proceedings of the 21st ACM conference on Hypertext and hypermedia*, pages 13–18. ACM, 2010.

[12] Sylvia Mercado Kierkegaard. How the cookies (almost) crumbled: Privacy & lobbyism. *Computer Law & Security Review*, 21(4):310–322, 2005.

[13] R. Kohavi. Mining e-commerce data: The good, the bad, and the ugly. In Foster Provost and Ramakrishnan Srikant, editors, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2035, page 2, 2001.

[14] R. E. Korf. Linear-space best-first search. *Artificial Intelligence*, 62(1):41–78, 1993.

[15] C-Y. Lin and F. J. Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 605, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

[16] Jonathan R. Mayer and John C. Mitchell. Third-party web tracking: Policy and technology. In *2012 IEEE Symposium on Security and Privacy*, pages 413–427. IEEE, 2012.

[17] Sharon Meraz. Is there an elite hold? traditional media to social media agenda setting influence in blog networks. *Journal of Computer-Mediated Communication*, 14(3):682–707, 2009.

[18] Michal Munk, Jozef Kapusta and Peter Švec. Data preprocessing evaluation for web log mining: reconstruction of activities of a web visitor. *Procedia Computer Science*, 1(1):2273–2280, 2010.

[19] Ashwin Paranjape, Robert West, Leila Zia and Jure Leskovec. Improving website hyperlink structure using server logs. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 615–624. ACM, 2016.

[20] Marko Poženel, Viljan Mahnič and Matjaž Kukar. Separating Interleaved HTTP Sessions Using a Stochastic Model. *Informatica*, 34:199–205, 2010.

[21] Marko Poženel, Viljan Mahnič and Matjaž Kukar. Separation of interleaved web sessions with heuristic search. In *2010 IEEE International Conference on Data Mining*, pages 411–420. IEEE, 2010.

[22] Aleksander Sadikov and Ivan Bratko. Pessimistic heuristics beat optimistic ones in real-time search. In *Proceedings of the 2006 Conference on ECAI 2006: 17th European Conference on Artificial Intelligence August 29 – September 1, 2006, Riva Del Garda, Italy*, volume 141, pages 148–152, Amsterdam, The Netherlands, The Netherlands, 2006. IOS Press.

[23] Aleksander Sadikov and Ivan Bratko. LRTA* works much better with pessimistic heuristics. In *Proceedings of the ECAI 2008: 18th European Conference on Artificial Intelligence*, pages 897–898, Amsterdam, The Netherlands, The Netherlands, 2008. IOS Press.

[24] C. Shahabi, F. Banaei-Kashani and J. Faruque. A framework for efficient and anonymous web usage mining based on client-side tracking. *Lecture Notes in Computer Science*, 2356:113–144, 2002.

[25] Edith G. Smit, Guda Van Noort and Hilde A. M. Voorveld. Understanding online behavioural advertising: User knowledge, privacy concerns and online coping behaviour in europe. *Computers in Human Behavior*, 32:15–22, 2014.

[26] M. Spiliopoulou, B. Mobasher, B. Berendt and M. Nakagawa. A framework for the evaluation of session reconstruction heuristics in Web-usage analysis. *INFORMS Journal on Computing*, 15(2):171–190, 2003.

[27] I. H. Ting, C. Kimble and D. Kudenko. A pattern restore method for restoring missing patterns in server side clickstream data. In *Web Technologies Research and Development - APWeb 2005*, volume 3399, pages 501–512. Springer-Verlag GmbH, March 2005.

[28] M. Viermetz, C. Stolz, V. Gedov and M. Skubacz. Relevance and impact of tabbed browsing behavior on web usage mining. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 262–269, Dec. 2006.

[29] J. Zhang and A. A. Ghorbani. The reconstruction of user sessions from a server log using improved time-oriented heuristics. In *Communication Networks and Services Research, 2004. Proceedings. Second Annual Conference on*, pages 315–322, May 2004.

Marko Poženel je asistent na Fakulteti za računalništvo in informatiko. Doktoriral je leta 2010 s področja računalništva. Njegovo področje raziskovanja vključuje tehnologije programske opreme, agilne metode razvoja programske opreme, podatkovno rudarjenje spletnih podatkov in analizo obnašanja uporabnikov na spletu.

Matjaž Kukar je izredni profesor na Fakulteti za računalništvo in informatiko Univerze v Ljubljani. Njegovi raziskovalni interesi vključujejo strojno učenje, odkrivanje zakonitosti v podatkih na splošno, analizo ROC, ocenjevanje zanesljivosti, odkrivanje zakonitosti v prostorsko-časovnih podatkih, tehnologije upravljanja velikih podatkov kot tudi aplikacije v medicinskem in poslovnem okolju.