

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS  
INFORMATIKOS KATEDRA

# **Šalto starto problemos rekomendacinėse sistemose sprendimas naudojant socialinių tinklų duomenis**

## **Applying Social Network Data for Cold Start Problem in Recommender Systems**

Magistro baigiamasis darbas

Atliko:	Andrius Juškevičius	(parašas)
Darbo vadovas:	lekt. Rimantas Kybartas	(parašas)
Recenzentas:	prof. habil. dr. Antanas Žilinskis	(parašas)

Vilnius – 2016

## Santrauka

Žmonės priimdami sprendimus dažnai pasikliauja draugų ir pažįstamų rekomendacijomis. Vienas iš rekomendacinių sistemų (toliau - RS) metodų - bendradarbiavimo filtravimas (angl. collaborative filtering, toliau BF) nors ir imituoja žmonių tarpusavio panašumą, negali identifikuoti, ką žmogus pažįsta, o ko ne. Socialinių tinklų duomenys užpildo šią spragą ir leidžia RS pateikti rekomendacijas atsižvelgiant ir į žmonių tarpusavio santykį.

Šiame darbe pateikta glausta rekomendacinių sistemų apžvalga, išnagrinėtas bendradarbiavimo filtravimo algoritmas, pristatyta šalto starto problema bei apžvelgtos socialinio tinklo duomenų taikymo galimybės sprendžiant šią problemą. Taip pat pasiūlyti trys nauji, socialinių tinklų duomenų panaudojimu besiremiantys metodai, kuriuos taikant galima spręsti šalto starto problemą.

**Raktiniai žodžiai:** rekomendacinė sistema, bendradarbiavimo filtravimas, socialinis tinklas, šaltas startas, pasitikėjimas

## Turinys

Ivadas .....	3
1. Litratūros apžvalga .....	5
1.1. Bendradarbiavimo filtravimas .....	5
1.1.1. Bendradarbiavimo filtravimo metodas .....	5
1.1.2. Šalto starto problema.....	7
1.1.3. Naudotojų panašumo apskaičiavimas .....	7
1.1.3.1. Pyrsono koreliacija .....	7
1.1.3.2. Apribota Pyrsono koreliacija .....	8
1.1.3.3. Spearmano rango koreliacija .....	8
1.1.3.4. Kosinuso panašumas .....	8
1.1.3.5. Euristinis PIP panašumo matas .....	8
1.1.3.6. Panašumas su svoriais .....	9
1.2. Socialiniai tinklai ir pasitikėjimu pagrįstos rekomendacinės sistemos .....	10
1.2.1. Socialiniai tinklai ir pasitikėjimo sąvoka.....	10
1.2.2. Pasitikėjimo apskaičiavimas .....	11
1.2.2.1. TidalTrust .....	11
1.2.2.2. MoleTrust .....	12
1.2.2.3. Pasitikėjimu pagrįstas svoris .....	13
2. Pasitikėjimu pagrįstos rekomendacinės sistemos modeliavimas ir siūlomi metodai .....	14
2.1. RS vertinimas .....	15
2.1.1. Vertinimo metrikos.....	15
2.1.2. Duomenų rinkinio skaidymas .....	16
2.2. Sričių panašumo metodas.....	17
2.2.1. Rekomendacinės sistemos su pasitikėjimu kategorijose modeliavimas .....	17
2.2.1.1. Kategorijos .....	17
2.2.1.2. Naudotojai .....	18
2.2.1.3. Elementai .....	19
2.2.1.4. Reitingai.....	19
2.2.1.5. Pasitikėjimai .....	20
2.2.1.6. Sugeneruoto duomenų rinkinio charakteristikos .....	21
2.2.2. Metodas.....	21
2.2.2.1. Pasitikėjimų pagrįstų RS metodai.....	24
2.2.3. Rezultatai .....	25
2.2.3.1. Eksperimentas naudojant RS su skirtingomis kategorijomis .....	25
2.2.3.2. Eksperimentas naudojant RS su panašiomis kategorijomis .....	30
2.3. Problemos ir iššūkiai.....	33
3. Išvados .....	35

## Ivadas

Kaskart, kai kažko ieškome, tiksliai patys nežinodami, ko - susiduriame su rekomendacijos poreikiu. Iš esmės, didžioji dalis dalykų apie kuriuos žinome, mums kažkada buvo viena ar kita forma pasiūlyta ar nurodyta. Taigi, didelė dalis pasaulio pažinimo proceso įvyksta rekomendacijų dėka. Rekomendacija, kaip reiškiny, gali įgyti įvairias, dažniausiai socialines, formas - informacijos galime gauti iš artimųjų arba tam tikrų atstovų (pavyzdžiui, finansų patarėjo arba konsultanto). Kita forma, apie kurią ir yra šis darbas, yra skaitmeninė - rekomendacinių sistemų (toliau - RS) generuojamos rekomendacijos skaitmeninėje erdvėje siekia palengvinti naudotojo patirtį renkantį jį dominančius elementus iš prieinamos aibės. Šios rekomendacijos gali ne tik palengvinti paieškos procesą, bet ir pasiūlyti bei sudominti naudotoją tokiais elementais, apie kuriuos naudotojas nė nenučiuotų. Šis bruožas yra ypač aktualus kitai šio santykio pusei - siūlytojui (pavyzdžiui, pardavėjui) dėl akivaizdžių priežasčių - jis tampa labiau matomas, žinomesnis, galų gale jis gali gauti materialinės naudos.

RS plačiai taikomos muzikos, kino ir elektroninės prekybos platformose. Vietoj įprastos paieškos šios sistemos siūlo elementus pasiremdamos naudotojų elgesio istorija. Vienas labiausiai naudojamų metodų - bendradarbiavimo filtravimas (angl. Collaborative Filtering, toliau - BF). Aibė sėkmingų interneto įmonių (pavyzdžiui, Amazon.com, Netflix.com, Last.fm) pritaikė BF metodus tam, kad padidinti naudotojų pasitenkinimą jų siūlomų produktų. Taikant BF daroma prielaida, kad istoriškai panašūs naudotojai išliks tokie ir ateityje. Taigi, esminė problema, kurią reikia spręsti - naudotojų panašumo vertinimas. Filtravimo procesas remiasi jau turimais duomenimis, kurie dėl problemos prigimties yra labai reti - sistemoje gali būti tūkstančiai naudotojų ir dar daugiau elementų, tačiau kiekvienas naudotojas dažniausiai būna įvertinęs tik labai mažą visų elementų dalį, taigi panašumo įvertinimas tampa iššūkiu. Nėgana to, kai sistemoje atsiranda naujas naudotojas, pradžioje apie jį žinoma per mažai, kad būtų galima pateikti patikimas rekomendacijas. Ši problema dar kitaip vadinama šalto starto (angl. cold start problem). Ji yra ypač svarbi ir dėl to, kad, jeigu naujas naudotojas per pakankamai trumpą laiką neįsitikins sistemos nauda, labai tikėtina, kad jis niekada ja nebesinaudos.

Ieškant šios problemos sprendimo būdų buvo atlikta nemažai tyrimų apie hibridines RS. Šių hibridinių RS esmė - taikant BF panaudoti informaciją apie elementų turinį. Turiniu pagrįstas RS nagrinėja atskira šaka, apie kurią šiame darbe nebus kalbama. Nors hibridinės RS ir išsprendžia daugelį problemų, tačiau turi vieną esminį trūkumą - hibridinė RS yra labai priklausoma nuo konteksto, kuriame ji naudojama, kitaip sakant, ji yra neuniversali. Be to, kai kurioms dalykinėms sritims yra labai sudėtinga apibūdinti naudotojo susidomėjimo elemento atributus, taigi neįmanoma sukurti

tokios RS.

Šio darbo tikslas – pasiūlyti metodą, kuriuo remiantis būtų galima išspręsti duomenų nepakankamumo problemą juos papildant duomenimis iš socialinių tinklų. Šie duomenys puikiai panaudojami pasitikėjimu pagrįstose RS. Pasitikėjimas gali būti traktuojamas kaip alternatyvus dydis panašumui. Šie du dydžiai skiriasi:

- pasitikėjimas nebūtinai yra išskaičiuojamas iš duomenų - jis gali būti išreikštas tiesiogiai.
- pasitikėjimas turi kryptį - tai yra naudotojas  $u_1$  gali pasitikėti  $u_2$  ne tiek pat, kiek  $u_2$   $u_1$ .

Pasitikėjimo tinklas - grafas, kurio viršūnės vaizduoja naudotojus, briaunos - santykius tarp jų, o briaunų svoriai - pasitikėjimo įverčius. Toks tinklas ir bus pamatas siūlomiems metodams, kaip spręsti šalto starto problemą, kai nepakanka duomenų naudotojų panašumui nustatyti.

Literatūros apžvalgoje suformuluoti bendradarbiavimo filtravimo naudotoju pagrįstu ir daiktu pagrįstu metodų apibrėžimai, pristatyta šalto starto problema ir aprašyti įvairių autorių pasiūlyti metodai šiai problemai spręsti. Tyrimai apie socialinių tinklų duomenų panaudojimą bus aptarti plačiau ir pristatyti jau atlikti darbai šia problemos sprendimo kryptimi. Taip pat gilinamasi į socialinių tinklų duomenų panaudojimo galimybes siekiant panaikinti (arba sušvelninti) šalto starto problemos efektą. Kitame skyriuje pristatytas būdas, kaip galima generuoti socialinių tinklų duomenis ir pasiūlyti trys nauji metodai naudojami RS su socialinių tinklų duomenimis - bendrų kaimynų metodas, atsižvelgiantis tik į ryšių egzistavimą tarp naudotojų, sričių panašumo metodas, kuris taikomas RS su kategorijomis, ir pasitikėjimo interpoliavimo metodas, kurio esmė - prognozuoti naudojo tarpusavio pasitikėjimą remiantis "paslėptais" RS duomenimis (juos naudojame generuodami RS duomenis). Trečiame skyriuje pateikta pasiektų rezultatų santrauka ir išvados.

# 1. Litratūros apžvalga

## 1.1. Bendradarbiavimo filtravimas

### 1.1.1. Bendradarbiavimo filtravimo metodas

Visų pirma, suformuluokime RS sprendžiamą problemą formaliai taip, kaip tai padaryta [2]. Vartotojų aibę pažymėkime  $U$  ir elementų aibę  $I$ . Be to, pažymėkime  $R$  aibę sistemoje turimų reitingų ir  $S$  – aibę galimų reikšmių, kurias gali įgyti reitingas (pvz.  $S = [1,5]$ ). Taip pat, tarkime, kad vienas reitingas  $r_{ui}$  gali būti priskirtas vienam elementui  $i \in I$  vieno naudotojo  $u \in U$ . Vartotojų poaibį, kuris yra įvertinęs elementą  $i$ , pažymėkime  $U_i$ . Analogiškai,  $I_u$  pažymėkime aibę elementų, kuriuos yra įvertinęs naudotojas  $u$ . Daiktų, kuriuos yra įvertinę abu naudotojai  $u$  ir  $v$ , aibę  $I_u I_v$  pažymėkime  $I_{uv}$ . Analogiškai,  $U_{ij}$  žymi aibę naudotojų, kurie yra įvertinę tiek elementą  $i$ , tiek  $j$ . Dvi dažniausiai sutinkamos problemos – geriausios ir geriausių  $N$  rekomendacijos problema. Vienas būdų spręsti šias problemas yra įvertinti funkciją  $f : U \times I \rightarrow S$ , kuri nuspėja reitingą  $f(u,i)$ . Ši funkcija tada yra naudojama naudotojo  $u_a$  rekomendacijai elemento  $i^*$ , kuriam įvertinamas reitingas turi didžiausią reikšmę  $i^* = \arg \max_{j \in I_{u_a}} f(u_a, j)$ . RS galima modeliuoti dviem būdais:

- Turiniu-pagrįstų metodų esmė – identifikuoti charakteristikas, kuriomis pasižymėjo elementai, kuriuos naudotojas įvertino palankiai praeityje ir tada naudotojui rekomenduoti kitus elementus su panašiomis charakteristikomis.
- Bendradarbiavimo-filtravimu pagrįsti metodai rekomenduoja elementus, kurie patiko naudotojams, turintiems panašias pirmenybes. BF metodai remiasi tik naudotojų suteiktais reitingais. Jie ieško panašumų tarp naudotojų pirmenybių ir tai lemia dvi geras savybes, kuriomis nepasižymi turiniu pagrįsti metodai
  - įžvalgumas - siūlomi ne tik akivaizdūs pasiūlymai, bet ir netikėti (t.y. tokie, kokių naudotojas kitomis aplinkybėmis turbūt nerastų)
  - pritaikymas skirtingose srityse, elementu pagrįstos rekomendacijos reikalauja specifinių srities parametrų duomenų (pvz., kiek tam tikras filmas yra komedija, kiek drama)

Bendradarbiavimo filtravimo sąvoką pirmąsyk panaudojo Goldberg [16]. Šis metodas remiasi artimiausių kaimynų metodu ir naudoja duomenis tiesiogiai generuojant rekomendacijas. Toliau darbe bus nagrinėjami būtent šiai klasei priklausantys metodai.

Bendradarbiavimo filtravimu pagrįsta reitingo prognozės esmė ta, kad parenkami artimiausi naudotojo kaimynai. Vartotojų tarpusavio artumas nustatomas naudojant panašumo metrikas, kurios bus aprašytos vėliau skyriuje 1.1.3. Šią prognozę galima atlikti dvejopai:

- Taikant artimiausių kaimynų regresiją, reitingas įvertinamas skaičiuojant pasvertą artimiausių kaimynų vidurkį.
- Taikant artimiausių kaimynų klasifikaciją, elemento reitingas parenkamas toks pats, kokį jam yra suteikęs artimiausias naudotojo kaimynas

Pagrindinis turiniu pagrįsto prieš naudotojų pagrįstą reitingo prognozavimo trūkumas yra tas, kad tokiu būdu sugeneruotos rekomendacijos yra nors ir tikslios, tačiau nelabai vertingos, nes rekomenduojami elementai pernelyg panašūs į tuos, kuriuos naudotojas jau žino. Šią problemą galima vertinti kaip pernelyg didelio pritaikymo (angl. over-specialization) problemą arba kaip išvalgumo (angl. serendipity) stygių. Be to, naudotojų pagrįstas metodas yra paremtas realiu žinių perdavimu iš lūpų į lūpas modeliu, todėl, tikėtina, geriau modeliuoja žinių išgavimą.

Norėdami prognozuoti naudotojo  $u$  reitingą elementui  $i$ , imame  $k$  artimiausių kaimynų  $N_i(u, k)$  ir ieškome jų vidurkio.

$$\hat{r}_{ui} = \frac{1}{N_i(u, k)} \sum_{v \in N_i(u, k)} r_{vi} \quad (1)$$

Ši formulė neatsižvelgia į naudotojų panašumą. Būtų neteisinga vertinti visus kaimynus vienodai, kai kurie yra panašūs į naudotoją  $u$ , o kai kurie visiškai nepanašūs. Čia įtraukiame svorių sąvoką. Svoriai gali reikšti arba panašumą (plačiau - 1.1.3), arba, kaip vėliau bus parodyta, vieno naudotojo pasitikėjimą kitu, apie kurį rašoma 1.2.2.

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i(u, k)} w_{uv} r_{vi}}{\sum_{v \in N_i(u, k)} |w_{uv}|} \quad (2)$$

Šioje formulėje naudojamas svertinis vidurkis yra dažniausiai praktikoje taikomas, paprastas ir tikslus būdas nustatyti prognozei, tačiau lieka klausimas - į kiek kaimynų reikia atsižvelgti. GroupLens sistemoje visi  $U \setminus \{u\}$  laikomi kaimynais; kitose sistemose kaimynai parenkami pagal panašumo slenkstį. Tinkamas kaimynų skaičiaus parinkimas leidžia įvertinti tikslesnes prognozes, nes taip sumažinamas kaimynų su maža koreliacija keliamas triukšmas. Dar kitas būdas - atsižvelgiant į dalykinę sritį parinkti konstantą. Geriausią kaimynų parinkimo strategiją galima išsiaiškinti tiesiog paeksperimentavus su konkrečiais duomenimis, nes įprastai RS viena nuo kitos labai skiriasi tiek dėl dalykinės srities subtilybių, tiek dėl RS dalyvaujančių naudotojų.

### 1.1.2. Šalto starto problema

Šalto starto problema susijusi su nepakankamu duomenų kiekiu. Šią problemą galima išskirti į dvi dalis:

- naudotojo šaltas startas
- elemento šaltas startas

Toliau bus rašoma tik apie naujo naudotojo problemą. Bendradarbiavimo filtravimu pagrįstuose metoduose, norint pateikti prasmingą rekomendaciją, visų pirma reikia suformuoti aiškų naudotojo pirmenybių vaizdą. Naujam naudotojui to padaryti faktiškai neįmanoma. Šia problemą galima spręsti visai negeneruojant rekomendacijų arba teikti rekomendacijas remiantis naudotojo profiliu - gyvenamąja vieta, amžiumi, lytimi ir panašiai. Dar kitas būdas - įvertinti trūkstamus duomenis - ir yra šio darbo esminis tyrimo objektas.

### 1.1.3. Naudotojų panašumo apskaičiavimas

Jau anksčiau buvo minėta, kad norint rasti prognozuojamą naudotojo  $u$  tam tikram elementui  $i$  suteikiamą reitingą, reikia žinoti svorius, kuriais matuojama kitų panašių naudotojų įtaka galutinei prognozei. Vienas šių svorių įvertinimo būdų - naudotojų panašumo išskaičiavimas iš reitingų matricos. Toliau pristatomi metodai, kurie padeda įvertinti naudotojų panašumą. Pyrsono, Spearmano koreliacija ir kosinuso panašumas detaliau aprašyti [2].

#### 1.1.3.1. Pyrsono koreliacija

Pyrsono koreliacija skirta statistinės koreliacijos radimui:

$$s(u,v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - \bar{r}_v)^2}} \quad (3)$$

Šis metodas susiduria su sunkumais, kai reikia paskaičiuoti panašumą tarp naudotojų, kurie bendrai yra įvertinę mažai elementų. Galima išeiti - nustatyti slenkstį, nuo kurio koreliacija būtų mažinama. Taigi panašumą  $s(u,v)$  tokiu atveju reiktų dauginti iš baudos funkcijos

$$\min\{|I_u \cap I_v|, 1\} \quad (4)$$



### 1.1.3.2. Atribota Pyrsono koreliacija

Kai kalbame apie šį metodą, pereiname nuo tolydinio prie kategorinio parametrų vertinimo. Be to, atsižvelgiama į nuokrypį ne nuo vidurkio, o nuo abejingumo įverčio. Jeigu turime reitingų skalę nuo 1 iki 7, tada 4 reiškia abejingumą. Pažymėkime  $r_x = 4$ . Tada Shardanand ir Maes pasiūlyta atribota Pyrsono koreliacija randama taip

$$s(u,v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)(r_{v,i} - r_z)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)^2} \sqrt{\sum_{i \in I_v \cap I_u} (r_{v,i} - r_z)^2}} \quad (5)$$

### 1.1.3.3. Spearmano rango koreliacija

Spearmano rango koreliacija panaši į Pyrsono koreliaciją, vienintelis skirtumas toks, kad skaičiuojant Spearmano koreliaciją, naudotojo reitingai yra surūšiuojami didėjimo tvarka, jiems priskiriami rangai - mažiausią reikšmę turintis reitingas gauna reikšmę 1. Tokiu būdu išvengiama reitingų normalizavimo problemos. Šis metodas veikia ne itin gerai, kai yra mažas galimų reikšmių skaičius, be to skaičiavimo požiūriu reikalaujantis daugiau resursų dėl surūšiavimo žingsnio.

### 1.1.3.4. Kosinuso panašumas

Šis metodas skiriasi nuo ankstesnių tuo, kad yra į problemą žiūrima ne iš statistinio, o iš tiesinės algebros požiūrio taško. Vartotojai atvaizduojami kaip  $|I|$  dimensijų turintys vektoriai, o panašumas apskaičiuojamas, kaip kosinuso atstumas tarp dviejų reitingo vektorių. Jis randamas sudauginant šiuos vektorius ir padalinant iš  $L2$  (Euklido) normų sandaugos:

$$s(u,v) = \frac{\mathbf{r}_u \cdot \mathbf{r}_v}{\|\mathbf{r}_u\|_2 \|\mathbf{r}_v\|_2} \quad (6)$$

### 1.1.3.5. Euristinis PIP panašumo matas

Euristinis panašumo matas pasiūlytas [7] kreipia dėmesį į šalto starto problemą. Dažniausias šalto starto problemos sprendimo būdas - naudoti hibridines RS, kurios naujiems naudotojams rekomendacijas pateikia naudodamos turinio informaciją ir tik surinkus pakankamai duomenų apie naudotoją, įjungiamas BF režimas. Ši panašumo metrika atsižvelgia į šalto starto problemą panašumą apskaičiuodama remdamasi trimis faktoriais - panašumu, poveikiu, populiarumu.

$$s(u_i, u_j) = \sum_{k \in C, j} PIP(r_{i,k}, r_{j,k}) \quad (7)$$

čia  $r_{ik}$  ir  $r_{jk}$  reitingai elementui  $k$  nuo naudotojų  $i$  ir  $j$  atitinkamai,  $PIP(r_{ik}, r_{jk})$  -  $PIP$  reikšmė reitingams  $r_{ik}$  ir  $r_{jk}$

$$PIP(r_1, r_2) = Proximity(r_1, r_2) \times Impact(r_1, r_2) \times Popularity(r_1, r_2) \quad (8)$$

Detalesnis aprašymas, kaip randamos šios reikšmės yra [7].

### 1.1.3.6. Panašumas su svoriais

[13] Said pastebėjo, kad dažniausiai naudojami panašumo matai (Pyrsono koreliacija, kosinuso panašumas) turi tokį trūkumą, kad jie neatsižvelgia į bendrai įvertintų elementų populiarumą - bendrai įvertinti populiarūs (įvertinti daugelio naudotojų) elementai vertinamam panašumui turėtų daryti mažesnę įtaką negu retai vertinami. Šį trūkumą siūloma spręsti panašumo matuose įvedant populiarumo svorius.

Tokiu būdu randama Pyrsono koreliacija atrodytų taip:

$$s_w(u, v) = \frac{\sum_{i \in I_u \cap I_v} w_i^s (r_{u,i} - \bar{r}_u) (r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} w_i^s (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} w_i^s (r_{v,i} - \bar{r}_v)^2}} \quad (9)$$

ir kosinuso panašumas:

$$s_w(u, v) = \frac{\sum_{i \in I_u \cap I_v} w_i^s \cdot r_{u,i} \cdot r_{v,i}}{\sqrt{\sum_{i \in I_u} w_i \cdot r_{u,i}^2} \sqrt{\sum_{i \in I_v} w_i^s \cdot r_{v,i}^2}} \quad (10)$$

o svoriai  $w_i^s$  gali randami būti randami tokiais būdais:

$$w_i^{s,inf} = \log \frac{|U|}{|U_i|} \quad (11)$$

$$w_i^{s,lin} = 1 - \frac{|U_i|}{|R|} \quad (12)$$

Čia  $|U|$  - naudotojų skaičius,  $|U_i|$  - naudotojų, įvertinusių elementą  $i$  skaičius,  $|R|$  reitingų skaičius.

Šaltinyje [13] parodyta, kad šis metodas geriausiai veikia vartotojams "po šalto starto" (angl. post cold start users), kai reitingų skaičius yra tarp 20 ir 80, kitiems režiams rezultatai buvo labai panašūs į tuos, kurie buvo gauti naudojant Pyrsono koreliaciją be svorių.

## 1.2. Socialiniai tinklai ir pasitikėjimu pagrįstos rekomendacinės sistemos

### 1.2.1. Socialiniai tinklai ir pasitikėjimo sąvoka

Socialinis tinklas - virtuali bendruomenė, kurios nariai bendrauja ir dalinasi tarpusavyje informacija. Žmonės tokiose bendruomenėse būna susiję - arba abipusiu (draugų), arba vienpusiu (pasekėjų) ryšių. Pasitikėjimu pagrįstų RS tikslas - įvertinti, kiek pasitikėjimo turi vienas naudotojas kitu, kai turimas pasitikėjimo tinklas (angl. web of trust). Įprastai toks įvertis randamas taikant propagavimo ir agregavimo operatorius. Propagavimo operatoriai nulemia, kaip bus elgiama su tranzityvumu. Kol kas nesigiliname į tai, kaip gaunami pasitikėjimo įverčiai, laikome juos duotais.

- Vienas dažniausiai naudojamų propagavimo operatorių (ypač, kai kalbame apie tikimybinį požiūrį) yra daugyba. Pavyzdžiui,  $u_1$  pasitiki  $u_2$  0.8, o  $u_2$  pasitiki  $u_3$  0.5, tada  $u_1$  pasitiki  $u_3$   $0.8 \times 0.5 = 0.4$ .
- Kitas operatorius - silpniausios grandies. Anksčiau pateikto pavyzdžio atveju  $u_1$  pasitikėjimas  $u_3$  būtų lygus 0.5.
- Konjunkcijos operatorius -  $\max(t_1 + t_2 - 1)$  ankstesniame pavyzdyje grąžintų 0.3  $A$  pasitikėjimą  $C$ .

Agregavimo operatoriai skirti susidoroti su situacijomis, kai yra keli propagavimo keliai. Šie operatoriai apjungia kelis pasitikėjimo įverčius į vieną. Žinoma, ne visi propagavimo keliai yra vienodo ilgio, tai yra, viename kelyje gali būti 1 naudotojas, kitame - 5. Verta pastebėti, kad svarbesni yra trumpesni keliai, ir kuo ilgesnis kelias - tuo mažiau informacijos jis suteikia. Taip yra dėl to, kad kiekvienas pasitikėjimo įvertis turi tam tikrą paklaidą - triukšmą, ir ilgesniame kelyje šio triukšmo yra daugiau. Ši problema nesunkiai sprendžiama taikant agregavimo operatorių. Galimi variantai - trumpiausio kelio operatorius, matematinis vidurkis, vidurkis su įvairiomis, atsižvelgiančiomis į kelio ilgį, schemomis.

Nors gali pasirodyti, kad nepasitikėjimas ir pasitikėjimas yra du dalykai priešinguose vienos tolydžios skalės galuose, tai yra tik kai kurių tyrėjų daroma prielaida, kuri leidžia supaprastinti problemą. Kitas, įgaunantis vis daugiau paramos, požiūris teigia, kad nepasitikėjimas negali būti prilyginamas pasitikėjimo nebuvimui.

Josang [18] kalba apie subjektyvią logiką (angl. subjective logic), kurioje, nepasitikėjimas yra traktuojamas kaip atskiras nuo pasitikėjimo dydis. Šios teorijos branduolys - subjektyvios nuomonės (angl. subjective opinions), kurios užrašomos taip:  $w_x^A = (b, d, u, a)$ , kur  $b$ ,  $d$  ir  $u$  apibūdina

pasitikėjimą, nepasitikėjimą ir neužtikrintumą. Pastebima, kad  $b, d, u \in [0, 1]$  ir  $b + d + u = 1$ . Parametras  $a \in [0, 1]$  nurodo, kokį svorį nustatant tikėtiną nuomonės įvertį (angl. opinion's probability expectation value) turi neužtikrintumas -  $E(w_x^A) = b + au$ . Šis modelis turi tikslius apibrėžimus ir formules, jomis galima manipuliuoti ir gauti analitiškai pagrindžiamus rezultatus, pavyzdžiui paaiškinti populiarumo bangas.

### 1.2.2. Pasitikėjimo apskaičiavimas

Pasitikėjimo tinkle dauguma naudotojų vienas kito nepažįsta. Nepaisant to, reikia nustatyti sąryšius tarp jų. Tam yra naudojamos pasitikėjimo metrikos, kurios remdamosi naudotojų santykiais nustato, kiek vienas naudotojas pasitiki kitu. Pasitikėjimo metrikos skyla į dvi klases.

- Lokalios metrikos įvertina pasitikėjimą kiekvienam naudotojui individualiai - dėl to jos gali būti tikslesnės ir reikalauja daugiau skaičiavimo resursų. Toliau bus pristatyti lokalių metrių pavyzdžiai - TidalTrust, MoleTrust.
- Globalios metrikos įvertina bendrą elemento reitingą visoje pasitikėjimo sistemoje. Apie jas toliau kalbama nebus, žymiausias pavyzdys - PageRank algoritmas naudojamas Google paieškos sistemoje.

Kaip minėta, pasitikėjimo skaičiavimui svarbi tranzityvumo prielaida, tačiau, ji teisinga tik tame pačiame kontekste - jeigu  $a$  pasitiki  $b$  kai kalbama apie automobilius, o  $b$  pasitiki  $c$  sodininkystės klausimais, nieko negalėsime pasakyti apie  $a$  pasitikėjimą  $c$  kompiuterijos žiniomis.

#### 1.2.2.1. TidalTrust

Ši formulė yra esminė Golbeck rekomendacijos algoritme. Algoritmo autoriai šią formulę išvedė atlikdami eilę eksperimentų, kurių metu jie ignoruodami tiesioginį naudotojo  $a$  pasitikėjimą naudotoju  $c$  tyrinėjo kelius, jungiančius šiuos du naudotojus. Lygindami taikant išskaidymą (angl. propagation) gautus įverčius su tikromis pasitikėjimo reikšmėmis jie pastebėjo, kad:

- trumpesni išskaidymo keliai leidžia apskaičiuoti tikslesnius pasitikėjimo įverčius
- keliai su didesnėmis pasitikėjimo reikšmėmis taip pat leidžia apskaičiuoti didesnius pasitikėjimo įverčius

Remiantis pirmu pastebėjimu buvo sugalvota, kad reikia apriboti kelio ilgį tarp naudotojų. Nustatant fiksuotą kelio ilgį gali atsitikti taip, kad tik maža dalis naudotojų gali būti pasiekiami. Dėl šios priežasties nustatytas kintamas galimas kelio ilgis - ilgiausias kelias, reikalingas sujungti tikslinį

naudotoją su naudotoju, įvertinusi elementą  $i$ .

Atsižvelgdami į kitą pastebėjimą (apie didesnes pasitikėjimo reikšmes vedančias prie tikslesnių įverčių) autoriai siūlo apriboti informaciją taip, kad ji būtų gaunama tik iš patikimiausių naudotojų. Tačiau čia vėl reikia pastebėti, kad skirtingi žmonės turi skirtingas pasitikėjimo skales - vienas gali pasitikėti visais, kitas - beveik niekuo. Be to, dažnai būna taip, kad mažai kelių turi tokią pačią pasitikėjimo reikšmę. Dėl šių priežasčių Golbeck nusprendė įvesti reikšmę, atspindinčią kelio stiprumą (t.y. mažiausią pasitikėjimo reitingą kelyje) ir apskaičiuoti maksimalų kelio stiprumą  $max$  (iš visų kelių, vedančių prie elementą vertinusių naudotojų), kuris po to naudojamas kaip slenkstis dalyvavimui algoritme.

$$t_{a,u} = \frac{\sum_{v \in WOT^+(a)} t_{a,v} t_{v,u}}{\sum_{v \in WOT^+(a)} t_{a,v}} \quad (13)$$

(13) pateikta TidalTrust formulė. Joje  $WOT^+(a)$  atspindi naudotojų aibę, kuriems naudotojo  $a$  pasitikėjimo jais reikšmė viršija slenkstį  $max$ .

Šis algoritmas yra rekursinis -  $t_{a,u}$  rekursiškai skaičiuojamas, kaip svertinis pasitikėjimo reikšmių  $t_{v,u}$  vidurkis. Šis algoritmas priklauso laipsniškų pasitikėjimo algoritmų klasei ir yra lokalaus pasitikėjimo metrikos pavyzdys.

Golbeck parodė, kad pasitikėjimu pagrįstas svertinis vidurkis kartu su TidalTrust nebūtinai visada yra pranašesnis už BF, tačiau duoda žymiai geresnius įverčius naudotojams, kurie nesutinka su vidutiniu elemento  $i$  reitingu.

#### 1.2.2.2. MoleTrust

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in R^T} t_{a,u} (r_{u,i} - \bar{r}_u)}{\sum_{u \in R^T} t_{a,u}} \quad (14)$$

(14) formulė - Massa [11] pasiūlyto rekomendacijų algoritmo pagrindas. Ši metrika susideda iš dviejų žingsnių:

- pirmame žingsnyje pašalinami pasitikėjimo tinkle esantys ciklai
- antrame žingsnyje atliekamas pasitikėjimo apskaičiavimas

Ciklų pašalinimo esmė ta, kad kiekvienas naudotojas tinkle būtų aplankytas tik kartą siekiant didesnio efektyvumo vykdant išskaidymą (angl. propagation). Ciklų pašalinimu transformuojame pradinį tinklą į kryptinį beciklį grafą. Tuomet pasitikėjimo prognozę  $t_{a,u}$  galime rasti atlikdami paprastą grafo apėjimą - visų pirma, randamas pasitikėjimas naudotojais, iki kurių atstumas lygus

1, tada pasitikėjimas tais, iki kurių atstumas 2 ir taip toliau. Verta pastebėti, kad pasitikėjimo naudotoju, esančių atstumu  $x$  priklauso nuo anksčiau apskaičiuotų pasitikėjimo reikšmių naudotojams esantiems atstumu  $x - 1$ .

Pasitikėjimas naudotojais, esančiais atstumu didesniu nei 1 skaičiuojamas panašiu būdu, kaip (13). TidalTrust naudotojas yra pridedamas prie  $WOT^+(a)$  tada ir tik tada, jeigu jis yra trumpiausiame kelyje nuo naudotojo  $a$  iki elemento  $i$ . MoleTrust atveju  $WOT^+(a)$  apima visus naudotojus, kurie įvertino tam tikrą elementą ir gali būti pasiekti pasitikėjimo tinklu per ne daugiau kaip  $d$  žingsnių. Parametras  $d$  vadinamas išskaidymo horizontu. Kitas MoleTrust parametras - pasitikėjimo slenkstis, kuris TidalTrust algoritme buvo apibrėžtas kaip dinamiška  $max$  reikšmė. MoleTrust pasitikėjimo slenkstis - fiksuotas dydis.

MoleTrust taip pat priklauso laipsniškų lokalių pasitikėjimo metrikų klasei. Algoritmo autoriai eksperimentu parodė, kad MoleTrust randa geresnius pasitikėjimo įverčius nei globalios pasitikėjimo metrikos, tokios kaip naudojamos pavyzdžiui eBay, ypač kai kalba eina apie kontroversiškus naudotojus, kuriuos dalis vertina kaip labai patikimus, o kita dalis - labai nepatikimus. Autoriai taip pat parodė, kad šis algoritmas išgauna tikslesnes prognozes naujiems naudotojams.

### 1.2.2.3. Pasitikėjimu pagrįstas svoris

Šis metodas pristatytas [12] naudoja vartotojo ir tiekėjo sąvokas. Reitingo prognozė skaičiuojama panašiai kaip (2):

$$c(i) = \bar{c} + \frac{\sum_{p \in P(i)} (p(i) - \bar{p})w(c,p,i)}{\sum_{p \in P(i)} |w(c,p,i)|} \quad (15)$$

$w(c,p,i)$  yra panašumo ir pasitikėjimo harmoninis vidurkis

$$w(c,p,i) = \frac{2(sim(c,p))(trust(p,i))}{sim(c,p) + trust(p,i)} \quad (16)$$

čia  $c$  - vartotojas (angl. consumer),  $p$  - gamintojas (angl. producer),  $i$  - elementas,  $sim(c,p)$  - panašumas tarp vartotojo ir gamintojo.  $trust(p,i)$  matuoja kiek  $c$  gali pasitikėti  $p$  elemento  $i$  vertinimu ir yra randamas taip:

$$trust(p,i) = \frac{|\{(c_k, i_k) \in CorrectSet(p) : i_k = i\}|}{|\{(c_k, i_k) \in RecSet(p) : i_k = i\}|} \quad (17)$$

Šis reiškinys rodo, kokia dalis naudotojo  $p$  rekomendacijų būna teisinga. Taip randamas pasitikėjimas vadinamas profilio lygio pasitikėjimu (angl. profile-level trust).

## 2. Pasitikėjimu pagrįstos rekomendacinės sistemos modeliavimas ir siūlomi metodai

Šio darbo tyrimo objektas - naujos tinklinių programų kartos atstovė - socialinė RS. Ji generuoja prognozes (rekomendacijas) apie naudotojams galinčius patikti elementus iš tam tikros, paprastai labai didelės aibės, remdamosi tarpusavio naudotojų santykiu. Sihna ir Swearingen [19] palygino RS ir draugų suteiktas rekomendacijas ir parodė, kad žmonės labiau pasitiki rekomendacijomis gautomis iš pažįstamų žmonių nei iš sistemos, veikiančios juodos dėžės (angl. black box) principu. Žinant, kad socialiniai tinklai vis populiarėja, o besinaudojančiųjų skaičius viršija milijardą, nesunku suprasti, kodėl RS kartu su socialiniais tinklais yra populiarus tyrimų objektas.

Tokiose sistemose naudotojas gauna rekomendaciją elemento, turinčio aukštą įvertinimą naudotojo WOT - pasitikėjimo tinkle (angl. web of trust). Pagrindiniai tokių sistemų įrankiai yra agregavimo (angl. aggregation) ir propagavimo (angl. propagation) operatoriai. Propagavimo operatorius taiko pasitikėjimo tranzityvumo prielaidą - jeigu naudotojas  $u_1$  pasitiki naudotoju  $u_2$ , o  $u_2$  pasitiki  $u_3$ , tai  $u_1$  pasitiki  $u_3$ . Agregavimo operatorius apjungia kelis pasitikėjimo įverčius į vieną.

Tikimybinio požiūriu pasitikėjimas gali įgyti tik dvi reikšmes - arba kitu naudotoju galima pasitikėti (su tikimybe  $p$ ), arba ne. Kitas, labiau įtikinantis ir panašesnis į realybę, yra laipsniškas požiūris, teigiantis, kad galima pasitikėti arba nepasitikėti tik iš dalies. Šiuo požiūriu pasitikėjimas nėra vertinamas kaip tikimybė, didesnė reikšmė tiesiog reiškia didesnę pasitikėjimą. Čia galima pastebėti ir analogiją su realiu gyvenimu - vienais žmonėmis pasitikime daugiau, kitais mažiau.

Tranzityvumo prielaida yra teisinga tik tame pačiame kontekste (toliau - srityje, kategorijoje). Jeigu  $u_1$  pasitiki  $u_2$  kai kalbama apie automobilius, o  $u_2$  pasitiki  $u_3$  sodininkystės klausimais, nieko negalėsime pasakyti apie  $u_1$  pasitikėjimą  $u_3$  kompiuterijos žiniomis.

Šiame darbe siūlomi metodai remiasi nauju duomenų aplinkos interpretavimu. Iki šiol buvo kalbėta apie sistemas, kuriose naudotojai turi kitiems naudotojams priskyrę tam tikrus skaitinius pasitikėjimo įverčius. Šiame darbe siūloma praplėsti šį apibrėžimą iki bendresnio atvejo, kuriame galimos kelios pasitikėjimo sritys, taigi vienas naudotojas kitam gali priskirti kelis įverčius pagal pasitikėjimo sritis, kitaip tariant, vienas vartotojas kitam priskiria pasitikėjimo vektorių. Taip pat, tinklo dalyviai gali būti tarpusavyje susiję ir be išreikšto pasitikėjimo įverčio, tai yra pasitikėjimas traktuojamas kaip neprivalomas esamo santykio atributas. Tada santykį tarp bet kurių  $u_1$  ir  $u_2$ , galime užrašyti kaip  $r_{u_1}(u_2) = (e_{u_1}(u_2), t_{u_1}(u_2))$ ,  $e_{u_1}(u_2) \in \{0,1\}$ ,  $t_{u_1}^k(u_2) \in [0,1]$ , kur  $k = 1, \dots, N$ , o  $N$  - pasitikėjimo sričių skaičius.  $e$  rodo ar tinklo dalyviai turi ryšį, o  $t_{u_1}(u_2)$  rodo naudotojo

$u_1$  pasitikėjimą naudotoju  $u_2$ , kuris, kai  $e = 0$ ,  $t_{u_1}(u_2) = \emptyset$ . Pačias pasitikėjimo sritis žymėsime  $T_1, T_2, \dots, T_N$ .

Šiame tyrime daroma prielaida, kad pasitikėjimo įverčius naudotojai vieni kitiems priskiria rankiniu būdu, remdamiesi savo subjektyvia nuomone apie kitų naudotojų patikimumą. Nors realioje sistemoje tokia prielaida, ko gero, nepasiteisintų, ši problema galėtų būti sprendžiama tyrime iš žmogaus ir kompiuterio sąveikos projektavimo požiūrio taško. Toks projektavimas, be abejo, priklausytų nuo aplinkos, kurioje norime įgalinti naudotojus išreikšti vienų kitais pasitikėjimą. Sprendimas galėtų būti pavyzdžiui toks:

- naudotojas atsidaro kito naudotojo apžvalgą
- sistema pastebi, kad naudotojas  $u_1$  skaito jau ne pirmą apžvalgą, kurią parašė  $u_2$
- sistema primena anksčiau skaitytas apžvalgas ir paklausia, kiek jis pritaria naudotojui  $u_2$
- jei naudotojas atsako, pasitikėjimo įvertis išsaugojamas

Kitas scenarijus yra, kai norime priskirti pasitikėjimą ne apžvalgininkui, o kitam asmeniui (pavyzdžiui, draugui). Tuomet galima tiesiog nueiti į to asmens anketą ir joje užpildyti pasitikėjimo įvertį (skalėje nuo 1 iki 5). Jeigu žinomas panašumo tarp naudotojų  $u$  ir  $v$  įvertis  $sim(u, v)$ , galima inicializuoti pasitikėjimą šiuo įverčiu ir esant progai paklausti naudotojo, ar jo pasitikėjimas naudotoju  $v$  yra lygus  $sim(u, v)$ . Toks metodas ypač aktualus, kai kalbama apie kelių pasitikėjimo sričių RS ir norime žinoti pasitikėjimus kiekvienoje jų. Šių ir kitų duomenų išgavimo būdų efektyvumo patvirtinimas arba paneigimas neįeina į šio darbo apimtį.

## 2.1. RS vertinimas

### 2.1.1. Vertinimo metrikos

Šio tyrimo tikslas – ištirti pasiūlytų metodų efektyvumą ir tikslumą sprendžiant šalto starto problemą rekomendacinėse sistemose. Tikslumas vertinamas naudojant "išimk vieną" metodą, kurio esmė tokia - iš duomenų išimamas vienas reitingas ir tada bandoma jį prognozuoti remiantis likusiais sistemos duomenimis. Tada vertinamas tikslumas ir padengimas. Tikslumas matuojamas taikant šias metrikas:

- $MAE$  - vidutinė absoliuti klaida (angl. mean absolute error) skaičiuoja visų prognozės klaidų vidurkį. Ši metrika ne visiškai atspindi RS tikslumą, nes taip pat vertina ir daug duomenų



turinčius ir šalto starto naudotojus.

$$MAE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} |\hat{r}_{ui} - r_{ui}|} \quad (18)$$

Kadangi mažai duomenų turintiems naudotojams tikslumas gali būti mažesnis, Massa ir Avesani pasiūlė kitą metriką, kuri suvienodina vieno naudotojo reikšmę vertinant vidutinę klaidą - vidutinę absoliučią naudotojo klaidą.

- *MAUE* - Vidutinė absoliuti naudotojo klaida (angl. mean absolute user error), kurią pasiūlė Massa ir Avesani skaičiuojama kiekvienam naudotojui atskirai, o tada randamas tų klaidų vidurkis. Ji skiriasi nuo *MAE* tuo, kad prognozės tikslumas kiekvienam naudotojui turi vienodą svorį, o *MAE* labiau atsižvelgia į aktyvesnius naudotojus
- *RMSE* - kvadratinė vidutinė klaida (angl. root mean squared error) - viena populiariausių metrikų, panaši į vidutinę absoliučią klaidą

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (\hat{r}_{ui} - r_{ui})^2} \quad (19)$$

Kitas vertinimo kriterijų grupė ypač svarbi, kai kalbama apie šaltą startą. Padengimą (angl. coverage) vertinsime dviem būdais:

- *RC* - reitingų padengimo esmė - palyginti reitingų, kuriuos algoritmas sugebėjo įvertinti taikant "išimk vieną" metodą, skaičių su visų sistemoje esančių reitingų skaičiumi
- *UC* - naudotojo padengimas lygina keliems naudotojams algoritmas sugebėjo prognozuoti bent vieną reitingą su skaičiumi naudotojų, kurie yra priskyrę reitingą bent vienam elementui.

Vertindami metodus skyriuje apie sričių panašumą naudosime tokį vertinimo kriterijų rinkinį - *MAE*, *MAUE*, *RMSE*, reitingų padengimą - *RC*, naudotojų padengimą - *UC*.

### 2.1.2. Duomenų rinkinio skaidymas

Tam, kad galėtume ištirti metodo efektyvumą skirtingiems naudotojų tipams. Išskiriame du įdomius naudotojų tipus:

- Šalto starto naudotojai - tie, kurie yra įvertinę mažiau nei 15 elementų.
- Ryžtingi naudotojai - tie, kurie turi daugiau reitingų, tačiau jie yra pasiskirstę plačiai apie vidurkį. Tokiais laikysime naudotojus, kurių reitingų standartinis nuokrypis didesnis nei 2.

## 2.2. Sričių panašumo metodas

### 2.2.1. Rekomendacinės sistemos su pasitikėjimu kategorijose modeliavimas

Šiuo metu nėra tokio duomenų rinkinio, tinkančio atliekamam tyrimui apie RS, kurioje elementai priklauso kategorijoms ir naudotojai išreiškia pasitikėjimą kategorijose. Dėl šios priežasties dalis tyrimo skirta RS modelio sudarymui ir duomenų generavimui. Siekiama sukurti duomenų struktūrą, turinčią tokius elementus:

- Kategorijos
- Naudotojai
- Elementai (vertinami produktai) priklausantys kategorijoms
- Naudotojo tarpusavio pasitikėjimai kategorijose (tolydi reikšmė tarp 0 ir 1)
- Naudotojų reitingai, priskirti elementams

Toliau bus aprašyti kiekvieno iš elementų generavimo algoritmai.

#### 2.2.1.1. Kategorijos

Kategorijų modeliavimas - pirmas algoritmo žingsnis. Juo siekiama apibrėžti ne tik kategorijas, kurioms gali priklausyti elementai bet ir kiekvieno elemento bruožus bei kiekvieno naudotojo pirmenybes. Šio generuojamo duomenų rinkinio duomenys gali būti vertinami kaip filmų rekomendacinės sistemos duomenys. Apibrėžiame pavyzdžiui tokias kategorijas:

- $X_1$  - drama
- $X_2$  - komedija
- $X_3$  - siaubo
- $X_4$  - trileris
- $X_5$  - fantastika

Toliau apibrėžiame, kiek kiekviena iš šių kategorijų yra susijusi su kitomis. Euristiškai sudarome matricą 1:

1 lentelė. Kategorijų matrica SP1

Kategorijos	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$x_1$	0.55	0.2	0.2	0.2	0.3
$x_2$	0.2	0.6	0.05	0.05	0.1
$x_3$	0.05	0.05	0.35	0.1	0.05
$x_4$	0.1	0.05	0.2	0.65	0.05
$x_5$	0.1	0.1	0.2	0	0.5

Čia  $X_1, \dots, X_5$  žymime kategorijas, o  $x_1, \dots, x_5$  kategorijas atitinkančius požymius (toliau - charakteristikas). Taigi iš šios matricos galime teigti, kad pavyzdžiui:

- $X_4$  (trileris) yra grynias žanras, tai yra, turintis daugiausiai savo kategoriją atitinkančio požymio (kadangi turi didžiausią matricos įstrižainėje esančią reikšmę)
- $X_3$  (siaubo) - mažiausiai gyna kategorija (nes bruožų pasiskirstymas yra tolygiausias)
- $X_4$  kategorija neturi  $x_5$  bruožo (trileris neturi fantastikos bruožų)

Akivaizdu, kad šie teiginiai yra subjektyvūs. Didesnio objektyvumo galima pasiekti, pavyzdžiui, sudarant kategorijų matricą remiantis apklausų duomenimis.

Atkreipkite dėmesį - tai tik pavyzdys, tyrimo eigoje bus atlikta eksperimentų su RS, kurios bus apibrėžtos kitomis kategorijų matricomis.

Ši matrica bus naudojama generuojant elementus. Nuo to, kokie yra elementai priklauso tai, kaip juos vertina naudotojai, o nuo to priklauso ir tai, kaip jie vertina vienas kito patikimumą. Taigi, ši matrica - RS duomenų generavimo pagrindas.

### 2.2.1.2. Naudotojai

Naudotojas apibrėžiamas kaip vektorius  $(y_1, y_2, y_3, y_4, y_5, q)$ , kur  $\sum_{i=1}^5 y_i = 1$  ir  $q \in [0,1]$ .  $y_1, y_2, y_3, y_4, y_5$  reiškia naudotojo pirmenybes - kiek svarbus jam yra tam tikras bruožas elemente.  $q$  - kokybės parametras rodo, kiek naudotojas yra jautrus elemento kokybei. Kokybės parametro motyvacija tokia - net jei žmogui apskritai nepatinka siaubo filmai, labai tikėtina, kad egzistuoja bent vienas kurį jis vertintų labai gerai (dėl to, kad tas filmas yra aukštos kokybės ir patinka daugumai žmonių).

Praktiškai algoritmas realizuojamas taip:

- sugeneruojame 5 atsitiktinius skaičius tarp 0 ir 1 (naudojant tolygų skirstinį)

- randame jų sumą
- kiekvienam bruožui priskiriame reikšmę lygią pirmame žingsnyje sugeneruotai reikšmei padalintai iš visų reikšmių sumos
- kokybės parametrai priskiriame atsitiktinę reikšmę tarp 0 ir 1

Taip užtikriname, kad naudotojai yra tikrai atsitiktiniai ir įvairūs pirmenybių prasme - naudotojui gali patikti tiek siaubo filmai, tiek komedijos, nors tarp šių kategorijų panašumo nėra.

### 2.2.1.3. Elementai

Elementas apibrėžiamas vektoriumi  $(c, z_1, z_2, z_3, z_4, z_5, q)$ . Čia  $c$  nurodo, kuriai kategorijai priklauso elementas, parametrai  $z_1, z_2, z_3, z_4, z_5$  rodo, kiek elementas pasižymi kiekvienu bruožu, o  $q$  - kokybės parametras. Generuojant elementus negalime taikyti tokio paties metodo, kaip naudotojo atveju, nes elementas priklauso tik vienai kategorijai, o tai reiškia, kad bruožų reikšmės negali būti visiškai atsitiktinės. Jas generuojame pasinaudodami normaliuoju skirstiniu su vidurkiu lygiu reikšmei gautai iš kategorijų matricos, aprašytos skyrelyje apie kategorijas ir parinktu standartiniu nuokrypiu (tokiu, kad duomenys būtų panašūs į realius - parinkus per didelę rezultatai gaunasi labai triukšmingi, šiame tyrime standartinį nuokrypį prilyginame konstantai lygiai 0.3). Vidurkis parenkamas taip: pažiūrėję į  $c$  reikšmę atfiltruojame kategorijų matricoje kategoriją (stulpelį). Tada turime vidurkių, naudojamų generuojant  $z_1, z_2, z_3, z_4, z_5$ , vektorių. Kokybės parametras, kaip ir naudotojo atveju, parenkamas atsitiktinai pagal normalųjį skirstinį su vidurkiu 0.6 ir standartiniu nuokrypiu lygiu 0.4. Jei sugeneruota reikšmė didesnė už 1 arba mažesnė už 0, ji priskiriama 1 arba 0 atitinkamai.

### 2.2.1.4. Reitingai

Naudotojo reitingai elementams generuojami naudojant jo pirmenybes ir reiklumo kokybei parametą bei atitinkamus produkto parametrus. Siekiama, kad jų pasiskirstymas būtų kuo artimesnis tikrovei, tai reiškia - nebūtų pasiskirstę galimų reikšmių kraštuose arba pernelyg vienodi. Sugeneruotų duomenų charakteristikos bus pateiktos kitame skyrelyje.

Kiekvienam naudotojui parenkamas atsitiktinis įvertintų elementų skaičius naudojant atsitiktinį dydį pasiskirsčiusį pagal normalųjį skirstinį su vidurkiu 30 ir standartiniu nuokrypiu lygiu 27. Parinktas didelis nuokrypis užtikrina, kad duomenys bus artimesni tikriems - Epinions.com duomenų rinkinyje vieno naudotojo įvertintų elementų skaičius svyruoja nuo 0 iki 655. Kiekvienam

atsitiktinai parinktam elementui generuojamas reitingas tokiu būdu:

$$r_u(p) = 5 \times ((1 - q_u) \sqrt{\text{pos}(\text{corr}(X_u, Y_p))} + q_u q_p) \quad (20)$$

čia

- $r_u(p)$  - naudotojo  $u$  reitingas elementui  $p$
- $q_u$  - naudotojo  $u$  kokybės reiklumo parametras
- $q_p$  - elemento  $p$  kokybės parametras
- $X_u$  - naudotojo  $u$  pirmenybių rinkinys
- $Y_p$  - elemento  $p$  bruožų rinkinys
- $\text{pos}(x) - f[-1,1] -> [0,1]$

Idealiais atvejais, kai naudotojo reiklumas kokybei ir elemento kokybė lygi 1 arba naudotojo reiklumas kokybei lygus 0, tačiau elemento charakteristikos tobulai atitinka naudotojo pirmenybes, reitingas lygus 5. Tyrimo eigoje pastebėta, kad koreliacijos funkcijos įtaka pernelyg maža, todėl ji padidinama naudojant pasirinktą iškilią funkciją (šiuo atveju šaknis suteikia pageidaujamą efektą).

### 2.2.1.5. Pasitikėjimai

Pasitikėjimo reikšmės - svarbiausios prognozuojant reitingus, parodančios kokį svorį suteikti patikėtinio nuomonei apie elementą. Šiame tyrime naudotojai vieni kitais pasitiki kategorijos lygmenyje. Buvo išbandyti du pasitikėjimo reikšmių generavimo būdai.

Taikant pirmąjį būdą pasitikėjimas tarp dviejų naudotojų tam tikroje kategorijoje generuojamas lyginant naudotojų tarpusavio pirmenybes tos kategorijos atžvilgiu. Taigi pasitikėjimas kategorijoje  $X_1$  tarp naudotojų  $u(0.1, 0.2, 0.2, 0.5, 0, q_u)$  ir  $v(0.2, 0.2, 0.2, 0.2, 0.2, q_v)$  randamas taip:

$$t_u(v) = \max(x_1^u, x_1^v) - \min(x_1^u, x_1^v) = 0.2 - 0.1 = 0.1 \quad (21)$$

Tokiu būdu rasti pasitikėjimai tenkina šias savybes:

- yra intervale tarp 0 ir 1
- nepriklauso nuo kategorijų skaičiaus

Tolimesnis tyrimas parodė, kad šis būdas nėra pakankamai geras. Pagrindinė to priežastis ta, kad vertinant pasitikėjimą tam tikroje kategorijoje naudojamas tik vienas (tą kategoriją atitinkantis) bruožas, o kategorijos savaime nėra vienaalytės - jos turi įvairių bruožų, kurie aprašyti kategorijų matricoje. Taigi, jei kategorijų matrica būtų vienetinė - šis būdas būtų efektyvesnis.

Kitas būdas geresnis - jis, nors ir netiesiogiai, atsižvelgia į kategorijų matricą. Naudotojų, kurie pasitiki vienas kitu, poros ir kategorijos, kurioms generuojamas pasitikėjimas parenkami atsitiktinai, kaip ir ankstesnio būdo atveju. Naudotojų porai pasitikėjimas generuojamas taip:

- parenkami  $n$  atsitiktinių elementų iš atitinkamos kategorijos ir jiems generuojami abiejų naudotojų reitingai (kaip aprašyta ankstesniame skyrelyje)
- turint abiejų naudotojų reitingų vektorius, galime rasti panašumą tarp jų taikant vieną iš panašumo metrikų
- rastas panašumas transformuojamas taip, kad priklausytų intervalui tarp 0 ir 1, o tada prilyginamas pasitikėjimui

Taikant tokį metodą atsižvelgiama į visas kategorijų charakteristikas. Tai labai svarbu tolimesniam tyrimui, ypač panašumo tarp sričių įvertinimui, kuris nagrinėjamas tolimesniuose skyriuose.

#### 2.2.1.6. Sugeneruoto duomenų rinkinio charakteristikos

Norint iliustruoti metodų veikimą skirtingoms RS, praverčia duomenų rinkiniai su skirtingomis charakteristikomis. Pavyzdžiui, norėdami parodyti propagavimo metodų efektyvumą kitame skyriuje, pravers duomenų rinkinys su mažai duomenų apie pasitikėjimą (tai yra naudotojai turės mažai patikėtinių). Tam, kad būtų aišku, koks duomenų rinkinys naudotas konkrečiu atveju, bus pateikta rinkinį apibūdinanti generavimo parametrų lentelė.

Elementų skaičius	300
Naudotojų skaičius	100
Naudotojo ryšių skaičiaus pasiskirstymas	N(10, 9)
Naudotojo įvertintų elementų skaičiaus pasiskirstymas	N(30,27)
Vertinamas bendrų elementų skaičius ieškant pasitikėjimo	12
Kategorijų matrica	SP1 1

#### 2.2.2. Metodas

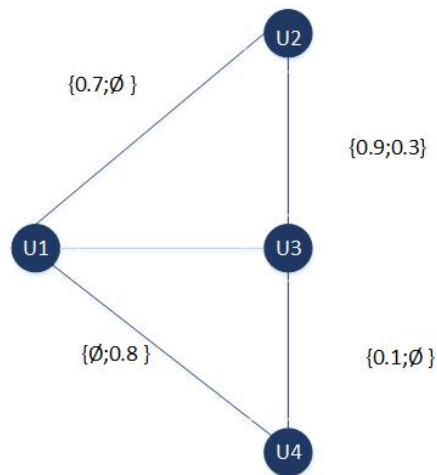
Šalto starto sąvoka nėra vienareikšmiškai apibrėžiama - negalime iš anksto žinoti, kiek ir kokių reikia duomenų, kad situacija tenkintų apibrėžimą ir taikomas metodas veiktų kaip tikimasi. Ap-

linkoje, apie kurią dabar rašoma, naudotojas gali būti šalto starto padėtyje, kai kalbame apie vieną sritį, tačiau kitoje srityje padėtis gali būti priešinga. Kitaip tariant, jeigu norime sužinoti, koks yra  $t_u^2(v)$ , ir žinome, kad naudotojo  $u$  pasitikėjimo naudotoju  $v$  srityje  $T_1$  lygis yra 0.8, o sričių  $T_1$  ir  $T_2$  panašumas  $\text{sim}(T_1, T_2) = 0.9$ , nieko nežinodami apie naudotojų santykį  $T_2$  klausimu, galime įvertinti jų tarpusavio pasitikėjimą tiesiog padauginę žinomos srities pasitikėjimo įvertį iš sričių tarpusavio panašumo įverčio.

Ši idėja yra pritaikoma ne tik šalto starto atveju, kai kalbama apie naudotoją, bet ir naujos srities šalto starto atveju. Socialinius tinklus pagal pasitikėjimo sričių daugialypiškumą galima išskirti į du tipus:

- daugiaprofilinius - juose galimos įvairios pasitikėjimo sritys - tokios, kurias galima surikiuoti pagal panašumą ir tarp pirmos bei paskutinės nėra jokio panašumo.
- specializuotos - juose pasitikėjimo sritys yra gana artimos. Tokio tinklo pavyzdys galėtų būti kino mėgėjų socialinis tinklas, o sritys - įvairūs žanrai.

Siūlomas metodas geriau veikia antrojo tipo atveju, kai sritys yra tarpusavyje panašios. Kai sritys pernelyg skirtingos, o panašumas mažas - panaudoti informaciją apie panašumą yra sudėtinga. Tarkime, kad turime situaciją pavaizduotą grafe 1, kuriame pateikti naudotojų tarpusavio pasitikė-



1 pav. Ryšių grafo fragmentas

jimai  $t_1, t_2$ , ir norime žinoti, kiek  $u_1$  pasitiki  $u_3$  srityje  $T_2$ . Tiesioginio kelio nėra, nes abiejuose galimuose keliuose -  $u_1 - u_2 - v$  ir  $u_1 - u_4 - v$  yra trūkstantų duomenų - pirmu atveju nežinome  $t_{u_1}^2(u_2)$ , antru -  $t_{u_4}^2(u_3)$ , tačiau matome, kad egzistuoja kelias  $u_1 - u_2 - u_3$ , pagal kurį galime įvertinti  $t_{u_1}^1(u_3)$

$$t_{u_1}^1(u_3) = t_{u_1}^1(u_2) \times t_{u_2}^1(u_3) = 0.7 \times 0.9 = 0.63$$

Žinodami, kad sričių panašumas  $\text{sim}(T_1, T_2) = 0.9$ , gauname

$$t_{u_1}^2(u_3) = t_{u_1}^1(u_3) \times \text{sim}(T_1, T_2) = 0.63 \times 0.9 = 0.6048$$

Toliau bus pasiūlytas algoritmas susidedantis iš dviejų etapų. Pirmas etapas skirtas panašumui tarp sričių radimui. Panašumas tarp sričių gali būti randamas globaliai - visai sistemai, arba kiekvienam naudotojui atskirai (jei tik naudotojas turi pakankamai duomenų). Šis metodas pavadintas TDS (angl. trust-based domain similarity). Antro etapo esmė - įvertinti dviejų naudotojų tarpusavio pasitikėjimą srityse, kuriose jo nėra. Čia pasiūlyti du metodai - *MAXDS* ir *AVGDS*.

Norėdami įvertinti panašumą tarp sričių turime turėti naudotojų porų ir jų tarpusavio pasitikėjimo pagal sritis sąrašą. Tada panašumą tarp sričių galime įvertinti taikydami vieną iš panašumo radimo metodų (Pyrsono, Spearmano koreliacija, kosinuso panašumas) turimiems pasitikėjimo (arba panašumo) duomenims.

---

**Algorithm 1** TDS metodas panašumo tarp sričių radimui

---

```

1: procedure GETCATEGORYSIMILARITY
2:   float[] trusts1;
3:   float[] trusts2;
4:   users  $\leftarrow$  GetAllUsers();
5:   foreach(var user in users):
6:     trustees  $\leftarrow$  user.GetTrusteesWithTrustInCategories(T1, T2);
7:     foreach(var trustee in trustees):
8:       trusts1.Add(trustee.T1);
9:       trusts1.Add(trustee.T2);
10:   similarity  $\leftarrow$  Correlation.Pearson(trusts1, trusts2);
11: end procedure

```

---

Tarkime, kad žinome panašumą tarp sričių. Belieka atsakyti į klausimą - kaip ši informacija gali padėti įvertinti pasitikėjimą tarp naudotojų. Tyrimo bus išbandyti du metodai:

- *MAXDS* metodas 2. Tarkime, kad turime naudotojų porą su žinomais pasitikėjimais dviejoje srityse ir nežinomais trijose. Norėdami įvertinti nežinomus pasitikėjimus, parenkame tą žinomą pasitikėjimo reikšmę, kuri yra didžiausia ir naudodami ją kaip pagrindą, nežinomas randame sudauginę ją su atitinkamos kategorijos panašumu. Šio metodo trūkumas tas, kad atsižvelgiama ne į visą žinomą informaciją.
- *AVGDS* metodu 3 siekiama panaudoti visą žinomą informaciją. Nežinomos pasitikėjimo reikšmės randamos ieškant randant žinomų pasitikėjimų sudaugintų su sričių panašumu vi-



durkį su svoriais. Svoriai šioje formulėje - tie patys vidurkiai.

$$t_u^{T_i}(v) = \frac{\sum_{j \in T} t_u^{T_j}(v) \times \text{sim}(T_i, T_j)^2}{\sum_{j \in T} \text{sim}(T_i, T_j)} \quad (22)$$

---

**Algorithm 2** MAXDS algoritmas trūkstamų pasitikėjimų tarp dviejų naudotojų radimui

---

```

1: procedure GETMISSINGTRUST
2:   threshold  $\leftarrow$  0.6;
3:   trusts  $\leftarrow$  GetTrusts(user1, user2);
4:   allCategories  $\leftarrow$  GetAllCategories();
5:   maxTrust  $\leftarrow$  (category, TrustValue);
6:   foreach(trust in trusts):
7:     If trust.TrustValue > maxTrust.TrustValue Then
8:       maxTrust  $\leftarrow$  trust;
9:     EndIf
10:  categoriesWithMissingTrust = allCategories.Except(trusts.categories);
11:  foreach(category in categoriesWithMissingTrust):
12:    categorySimilarity  $\leftarrow$  GetCategorySimilarity(maxTrust.category, category);
13:    newTrust  $\leftarrow$  (category, maxTrust  $\times$  categorySimilarity);
14:    If newTrust  $\geq$  threshold Then
15:      newTrust.Save();
16:    EndIf
17: end procedure

```

---

Kitas būdas, remiantis kuriuo galime rasti panašumą tarp sričių - ieškoti panašumo tarp sričių charakteristikų iš kategorijų matricos. Tai realizuojama tiesiog taikant bet kurią panašumo metriką kategorijų charakteristikų vektoriams. Nors metodą sunku pritaikyti realiems RS duomenims, nes kategorijų matrica nėra žinoma, jis įdomus teoriniu požiūriu. Tiesa, šis būdas veikia tik vertinant sričių panašumą visoje sistemoje.

### 2.2.2.1. Pasitikėjimų pagrįstų RS metodai

Kai kalbama apie rekomendacines sistemas su socialinių tinklų duomenimis daugiausia tyrimų [kokių] atlikta nagrinėjant agregavimo ir propagavimo metodus, kurie remiasi prielaida apie pasitikėjimo tranzityvumą. Šio tyrimo kontekste į šiuos metodus galima žiūrėti kaip į tam tikrą duomenų papildymo būdą prieš taikant sričių panašumo metodą.

Tyrimė bus išbandyti keli trumpiausio kelio šeimos algoritmai. Trumpiausio kelio algoritmas randa trumpiausią kelią tarp dviejų naudotojų ir randa pasitikėjimą tarp jų vienu iš šių operacijų pasitikėjimo įverčiams, esantiems pasitikėjimo kelyje:

- daugyba - SHORTMULTI

---

**Algorithm 3** AVGDS algoritmas trūkstančių pasitikėjimų tarp dviejų naudotojų radimui

---

```
1: procedure GETMISSINGTRUST
2:   threshold  $\leftarrow$  0.6;
3:   trusts  $\leftarrow$  GetTrusts(user1, user2);
4:   allCategories  $\leftarrow$  GetAllCategories();
5:   maxTrust  $\leftarrow$  (category, TrustValue);
6:   categoriesWithMissingTrust = allCategories.Except(trusts.categories);
7:   foreach(category in categoriesWithMissingTrust):
8:     numerator  $\leftarrow$  0
9:     denominator  $\leftarrow$  0
10:    foreach(trust in trusts):
11:      categorySimilarity  $\leftarrow$  GetCategorySimilarity(trust.category, category);
12:      numerator  $\leftarrow$  numerator + categorySimilarity  $\times$  categorySimilarity  $\times$ 
        trust.TrustValue;
13:      denominator  $\leftarrow$  denominator + categorySimilarity;
14:      newTrust  $\leftarrow$  numerator / denominator;
15:      If newTrust  $\geq$  threshold Then
16:        newTrust.Save();
17:      EndIf
18: end procedure
```

---

- aritmetinis vidurkis - SHORTARI
- harmoninis vidurkis - SHORTHARM

### 2.2.3. Rezultatai

Šiuo eksperimentu siekiama ištirti sričių panašumo metodo efektyvumą. Tačiau be šio galutinio tikslo, taip pat galima paminėti ir tarpinius tikslus - išbandyti duomenų rinkinio generavimo algoritmą ir kitų žinomų metodų, naudojamų pasitikėjimų radimui, efektyvumą.

Eksperimentas atliktas naudojant du skirtingus generuotus duomenų rinkinius. Pirmas - duomenų rinkinys, aprašytas ankstesniame skyriuje 2.2.1. Jis pasižymi tuo, kad kategorijos yra gana skirtingos ir sričių panašumas yra mažas. Kitas duomenų rinkinys bus generuojamas naudojant kategorijų matricą, kurioje sritys yra tarpusavyje panašios.

#### 2.2.3.1. Eksperimentas naudojant RS su skirtingomis kategorijomis

Ankstesniame skyrelyje sugeneruoto duomenų rinkinio charakteristikos yra tokios:

Naudotojų, įvertinusių bent vieną elementą, skaičius	87
Šalto starto naudotojų skaičius	18
Ryžtingų naudotojų skaičius	6
Reitingų standartinis nuokrypis	1.5

Reitingų pasiskirstymas:

1	2	3	4	5
612	345	370	730	839

Šiam duomenų rinkiniui pagal nustatytus vertinimo kriterijus gauname tokius rezultatus:

2 lentelė. BF rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	Naudotojų skaičius	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	<i>RC</i>	<i>UC</i>
Visi naudotojai	87	1.04	0.98	1.5	0.22	0.65
Šalto starto naudotojai	19	-	-	-	0	0
Ryžtingi naudotojai	6	1.17	1.17	1.84	0.06	0.17

Tokiems duomenims galime taikyti globalaus sričių panašumo metodą. Išbandyti AVGDS 3 (sričių panašumo vidurkio) ir MAXDS 2 (sričių panašumo maksimumo) metodai.

3 lentelė. BF + AVGDS rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	Naudotojų skaičius	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	<i>RC</i>	<i>UC</i>
Visi naudotojai	87	1.04	0.98	1.49	0.33	0.69
Šalto starto naudotojai	19	-	-	-	0	0
Ryžtingi naudotojai	6	0.96	0.67	1.43	0.11	0.5

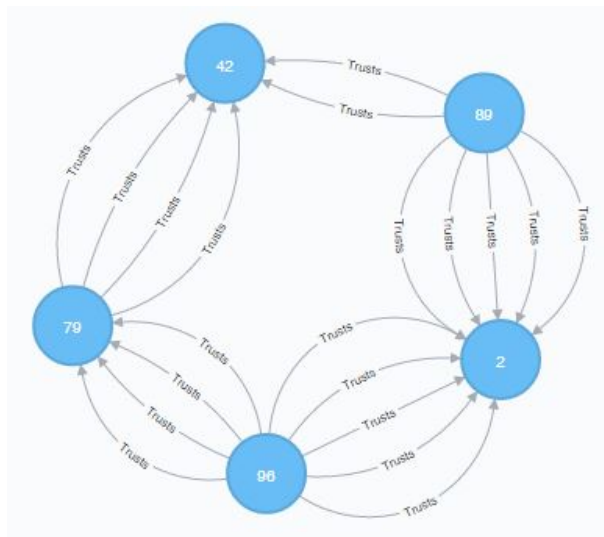
4 lentelė. BF + MAXDS rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	Naudotojų skaičius	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	<i>RC</i>	<i>UC</i>
Visi naudotojai	87	1.03	0.95	1.46	0.33	0.69
Šalto starto naudotojai	19	-	-	-	0	0
Ryžtingi naudotojai	6	1.03	0.69	1.59	0.11	0.5

Iš rezultatų matosi, kad tiek tikslumas, tiek padengimas pagerėjo visų naudotojų imčiai. Didelis tikslumo padidėjimas ryžtingų naudotojų atveju iš dalies gali būti paaiškintas atsitiktinumu dėl mažos imties, tačiau atlikus daugiau eksperimentų pastebėta, kad tikslumas beveik visada nežymiai keičiasi į gerąją pusę, o reitingų padengimas didėja vidutiniškai apie 50% kiekvienai imčiai. 2 pav. matosi, kaip atrodo ryšių grafas po sričių panašumo metodo pritaikymo.

Nepastebėta reikšmingo skirtumo tarp rezultatų, gautų taikant *MAXDS* ir *AVGDS* metodus, dėl to toliau bus taikomas tik *AVGDS* metodas (nes jis atsižvelgia į daugiau informacijos).

Sričių panašumo metodas nekuria naujų ryšių tarp naudotojų, kurie nieko vienas apie kitą nežino - tą daro kiti metodai, taikantys propagavimo ir agregavimo operatorius. Tačiau šiuos du metodų tipus galima kombinuoti ir taikyti kartu.



2 pav. Ryšių grafo fragmentas pritaikius sričių panašumo metodą

Šis duomenų rinkinys buvo sugeneruotas parinkus tokius parametrus, kad jame egzistuotų duomenų retumo problema. Tai matome iš nedidelių  $RC$  ir  $UC$  reikšmių. Egzistuojantis problemos sprendimo būdas - taikyti metodus, vertinančius naudotojų tarpusavio pasitikėjimą. Tam pačiam duomenų rinkiniui išbandyti trys trumpiausio kelio metodai, aprašyti ankstesniame skyrelyje (SHORTMULTI, SHORTARI, SHORTGEO). Siekiant vertinti tik svarbias pasitikėjimo reikšmes, nustatome slenkstį lygų 0.9, nuo kurio saugosime rastus pasitikėjimus.

5 lentelė. BF + SHORTMULTI rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	$MAE$	$MAUE$	$RMSE$	$RC$	$UC$
Visi naudotojai	1.05	0.99	1.62	0.51	0.87
Šalto starto naudotojai	1	0.83	1.55	0.23	0.55
Ryžtingi naudotojai	1.06	1.03	1.75	0.55	1

6 lentelė. BF + SHORTARI rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	$MAE$	$MAUE$	$RMSE$	$RC$	$UC$
Visi naudotojai	1.15	1.09	1.88	0.65	0.92
Šalto starto naudotojai	1.10	1.04	1.81	0.35	0.67
Ryžtingi naudotojai	1.27	1.31	2.33	0.65	1

Naudojant geometrinį vidurkį išvesti pasitikėjimo įverčiai neviršija 0.9, dėl to parenkame kitokį slenkstį - 0.6.

7 lentelė. BF + SHORTGEO rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	<i>RC</i>	<i>UC</i>
Visi naudotojai	1.16	1.14	1.84	0.8	0.98
Šalto starto naudotojai	1.16	1.1	1.88	0.52	0.89
Ryžtingi naudotojai	1.27	1.30	2.27	0.78	1

Atlikus eksperimentus paaiškėjo, kad geriausiai turimiems duomenims veikia SHORTMULTI metodas. Tai paaiškinama tuo, kad jis atsižvelgia į pasitikėjimo mažėjimą (angl. trust decay) esant ilgesniems pasitikėjimo keliams. SHORTARI ir SHORTGEO labiau padidina padengimą, tačiau tikslumas sumažėja pernelyg smarkiai, kad šie metodai būtų vertingi praktikoje.

Kaip minėta anksčiau, sričių panašumo metodas gali būti taikomas nepriklausomai nuo metodų, prognozuojančių naudotojų tarpusavio pasitikėjimą naudojant propagavimo ir agregavimo operatorius. Jau parodyta, kaip globalus sričių panašumas veikia su baziniais RS duomenimis. Dabar bus siekiama ištirti, kaip veikia sričių panašumo ir agregavimo bei propagavimo metodų kombinacijos. Jau parodyta, kad geriausiai iš tiriamų agregavimo ir propagavimo metodų veikia SHORTMULTI metodas, todėl toliau bus tiriamos trys kombinacijos:

- SHORTMULTI + AVGDS naudojant sričių panašumus gautus taikant TDS metodą visiems naudotojams. Sričių panašumo reikšmės pateiktos 8 lentelėje.
- SHORTMULTI + AVGDS sričių panašumus randamas kiekvienam naudotojui ir jo pasitikėjimai išskaičiuojant naudojant jo asmeninį sričių panašumo suvokimą.
- SHORTMULTI + AVGDS naudojant sričių panašumus gautus skaičiuojant panašumą tarp sričių charakteristikų apibrėžtų kategorijų matricoje. Sričių panašumo reikšmės pateiktos 9 lentelėje.

8 lentelė. Sričių panašumo matrica taikant TDS metodą

Kategorijos	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$X_1$	1	0.62	0.37	0.48	0.63
$X_2$	0.62	1	0.08	0.31	0.43
$X_3$	0.37	0.08	1	0.53	0.45
$X_4$	0.48	0.30	0.53	1	0.26
$X_5$	0.63	0.43	0.46	0.26	1

9 lentelė. Panašumo tarp kategorijų matrica

Kategorijos	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$X_1$	1	0.77	0.8	0.32	0.56
$X_2$	0.78	1	0.77	0.24	0.43
$X_3$	0.88	0.77	1	0.35	0.56
$X_4$	0.32	0.24	0.35	1	0.47
$X_5$	0.56	0.43	0.6	0.47	1

Naudojant 8 panašumo matricą trūkstamai informacijai apie pasitikėjimą užpildyti ir naudojant pasitikėjimo slenkstį 0.6 gaunami tokie rezultatai.

10 lentelė. BF + AVGDS (su sričių panašumais, gautais naudojant kategorijų matricą) rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.17	1.12	1.95	0.61	0.89
Šalto starto naudotojai	1.08	1.06	1.69	0.35	0.61
Ryžtingi naudotojai	1.16	1.13	2	0.76	1

Naudojant 9 panašumo matricą trūkstamai informacijai apie pasitikėjimą užpildyti kartu su pasitikėjimo slenksčiu lygiu 0.6 gaunami tokie rezultatai.

11 lentelė. SHORTMULTI + AVGDS (su sričių panašumais, gautais naudojant TDS), su slenksčiu 0.6, rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.07	0.96	1.56	0.55	0.87
Šalto starto naudotojai	1.02	1.05	1.79	0.3	0.56
Ryžtingi naudotojai	1.09	1.02	1.67	0.54	UC

12 lentelė. SHORTMULTI + AVGDS (su sričių panašumais, gautais naudojant TDS), su slenksčiu 0.3, rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.12	1.12	1.8	0.76	0.88
Šalto starto naudotojai	1.07	1.12	1.72	0.57	0.61
Ryžtingi naudotojai	1.13	1.12	1.95	0.9	1

Pastebime, kad turint tokias panašumo reikšmes slenksčio reikšmė lygi 0.6 yra labai didelė - iš tiesų taikydami šį metodą papildomos informacijos galime gauti tik apie  $X_1$  ir  $X_2$  bei  $X_1$  ir  $X_5$  kategorijų panašumus (nes tik jų sandauga su žinomu pasitikėjimu, mažesniu už 1, gali viršyti 0.6). Metodas taip pat buvo išbandytas su slenksčiu lygiu 0.3. Nors padengimas ir padidėjo, tačiau tikslumas

sumažėjo atitinkamai.

Paskutinis metodas, kuris bus išbandytas su šiuo duomenų rinkiniu - sričių panašumo naudotojo lygmenyje. Kiekvienam naudotojui rasime jo asmeninę sričių panašumo matricą.

13 lentelė. SHORTMULTI + AVGDS (sričių panašumai gauti taikant TDS naudotojo lygmenyje), su slenksčiu 0.6, rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.15	1.05	1.91	0.59	0.87
Šalto starto naudotojai	1.10	0.86	1.88	0.24	0.56
Ryžtingi naudotojai	1.14	1.12	1.95	0.9	1

Šio metodo rezultatai ne tokie geri, kaip būtų galima tikėtis - šalto starto naudotojams tiek padengimas, tiek tikslumas gaunamas geresnis taikant globalų sričių panašumą. Tai galima paaiškinti tuo, kad ieškant naudotojo asmeninio sričių panašumo atsižvelgiama į per mažai duomenų ir dėl to vertinimas būna labiau atsitiktinis.

Paskutinė metodų kombinacija - AVGDS + SHORTMULTI, pasirodo, veikia geriausiai.

14 lentelė. AVGDS + SHORTMULTI (sričių panašumai gauti taikant TDS naudotojo lygmenyje), su slenksčiu 0.6, rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.01	0.99	1.46	0.43	0.86
Šalto starto naudotojai	0.87	0.96	1.88	0.2	0.5
Ryžtingi naudotojai	1.03	0.98	1.75	0.47	1

Tokius rezultatus galima paaiškinti tuo, kad sričių panašumo metodas, taikomas po SHORTMULTI metodo, randa iškreiptus panašumo tarp sričių įverčius. Kita vertus, pirma taikant sričių panašumo metodą, papildome duomenis "teisingesniais" duomenimis, kuriems po to taikomas SHORTMULTI, generuoja geriausius šiame eksperimente pasiektus rezultatus.

Sunku vertinti gautus rezultatus vienareikšmiškai, nes jie susideda iš kelių dydžių. Vis dėlto, vertinant rezultatus, gautus taikant sričių panašumo metodą, reikia pasakyti, kad geriausi rezultatai gauti naudojant sričių panašumus išskaičiuotus iš kategorijų matricos. Viena vertus, tai reiškia, kad kiti metodai, kuriuos galima būtų pritaikyti realioms duomenims - (tiek globalus, tiek naudotojo lygmens), neveikia taip gerai kaip galėtų. Iš kitos pusės, tai įrodo, kad parinkus tinkamas panašumo reikšmes galima išgauti gerų rezultatų siekiant išspręsti duomenų retumo problemą.

### 2.2.3.2. Eksperimentas naudojant RS su panašiomis kategorijomis

Analogiškas eksperimentas buvo atliktas kitokiai RS, kurioje kategorijos yra panašios. Šiame skyriuje naudojamas duomenų rinkinys apibrėžiamas parametru rinkiniu 15.

15 lentelė. RS duomenų rinkinio generavimo parametrai

Elementų skaičius	300
Naudotojų skaičius	100
Naudotojo ryšių skaičiaus pasiskirstymas	N(10, 9)
Naudotojo įvertintų elementų skaičiaus pasiskirstymas	N(30,27)
Vertinamas bendrų elementų skaičius ieškant pasitikėjimo	12
Kategorijų matrica	SP2 16

16 lentelė. Kategorijų matrica SP2

Kategorijos	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$x_1$	0.6	0.5	0.4	0.6	0.3
$x_2$	0.2	0.1	0.2	0	0.3
$x_3$	0.1	0.1	0.1	0.1	0.1
$x_4$	0.1	0.1	0.1	0.2	0.1
$x_5$	0	0.1	0.2	0.1	0.2

Sugeneruoto duomenų rinkinio 16 charakteristikos yra tokios:

Naudotojų, įvertinusių bent vieną elementą, skaičius	87
Šalto starto naudotojų skaičius	18
Ryžtingų naudotojų skaičius	6
Reitingų standartinis nuokrypis	1.47

Reitingų pasiskirstymas:

1	2	3	4	5
532	339	373	786	849

Šiam duomenų rinkiniui pagal nustatytus vertinimo kriterijus gauname tokius rezultatus:

17 lentelė. BF rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	$MAE$	$MAUE$	$RMSE$	RC	UC
Visi naudotojai	1.04	0.91	1.51	0.21	0.63
Šalto starto naudotojai	-	-	-	0	0
Ryžtingi naudotojai	1.41	1.41	2.40	0.09	0.17

Pritaikę AVGDS (naudojant sričių panašumus, gautus taikant TDS) su slenksčiu 0.3 baziniams duomenims gauname tokią sričių panašumo matricą:



18 lentelė. Panašumo tarp kategorijų matrica

Kategorijos	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$
$X_1$	1	0.58	0.76	0.24	0.76
$X_2$	0.58	1	0.66	0.23	0.30
$X_3$	0.76	0.66	1	0.20	0.60
$X_4$	0.24	0.23	0.20	1	0.10
$X_5$	0.76	0.30	0.60	0.10	1

Gauti rezultatai:

19 lentelė. BF rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	$MAE$	$MAUE$	$RMSE$	RC	UC
Visi naudotojai	1.07	1	1.57	0.34	0.69
Šalto starto naudotojai	-	-	-	0	0
Ryžtingi naudotojai	1.30	1.30	2.23	0.14	0.17

Vėl matome, kad padengimui padidėjus, tikslumas šiek tiek sumažėja. Metodas prie 743 sistemoje esančių originalių pasitikėjimų, pridėjo dar 590 esamiems vartotojams.

Iš ankstesnio skyrelio jau aišku, kad iš tiriamų propagavimo ir agregavimo metodų geriausiai veikia SHORTMULTI metodas. Šiam duomenų rinkiniui šio metodo rezultatai panašūs. Kur kas įdomesni rezultatai gaunami taikant SHORTMULTI metodą duomenų rinkiniu, kuriam jau buvo pritaikytas sričių panašumo metodas

20 lentelė. AVGDS + SHORTMULTI rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	$MAE$	$MAUE$	$RMSE$	RC	UC
Visi naudotojai	0.97	0.86	1.40	0.34	0.85
Šalto starto naudotojai	0.76	0.68	0.95	0.15	0.55
Ryžtingi naudotojai	0.96	0.4	1.61	0.38	0.67

Matome, kad SHORTMULTI metodas taikomas po to, kai buvo pritaikytas sričių panašumo metodas duoda daug geresnį tikslumą ir ne prastesnį padengimą nei kitais atvejais. Dar labiau jį galima padidinti vėl pritaikius sričių panašumo metodą. Verta paminėti, kad duomenims kuriems jau buvo pritaikytas AVGDS + SHORTMULTI metodas sričių panašumų matrica, randama taikant TDS gaunama jau stipriai iškreipta - panašumai tarp sričių tapo artimesni 0. Gaunamas kiek prastesnis tikslumas, tačiau reitingų padengimas dar labiau padidėjo.

21 lentelė. AVGDS + SHORTMULTI + AVGDS rezultatai taikomi RS duomenims

Duomenų rinkinio poaibis	<i>MAE</i>	<i>MAUE</i>	<i>RMSE</i>	RC	UC
Visi naudotojai	1.02	0.97	1.40	1.51	0.85
Šalto starto naudotojai	1	0.89	1.54	0.24	0.55
Ryžtingi naudotojai	0.98	0.87	1.68	0.51	0.67

Lyginant metodų veikimą RS su panašiomis ir skirtingomis kategorijomis tikėtasi, kad išbandyti metodai tikslumo prasme veiks geriau RS su panašiomis kategorijomis. Šis spėjimas pasitvirtino - kalbant apie geriausią metodų kombinaciją - AVGDS + SHORTMULTI - RS su panašiomis sritimis buvo fiksuojamas didesnis tikslumas. Tačiau verta paminėti, kad skirtumas nėra toks didelis (*MAE* rezultatas visiems naudotojams - 1.01 ir 0.97).

Nors šiame tyrime nebuvo nagrinėjama metodų greitimeika - sričių panašumo metodas yra labai lengvai realizuojamas ir resursų reiklumo prasme gana lengvas. To negalima pasakyti apie propagavimo ir agregavimo metodus - net pats paprasčiausias trumpiausio kelio metodas 300 naudotojų trunka apie 5 minutes algoritmą vykdant paprastu namų kompiuteriu.

## 2.3. Problemos ir iššūkiai

Didžiausia problema šio tyrimo srityje yra realių duomenų nebuvimas ir negalėjimas praktiškai įvertinti šių metodų tinkamumo. Nėra žinomo socialinio tinklo, kuriame naudotojai išreikštų pasitikėjimą vienas kitu tolydžioje skalėje ir pasitikėjimai galėtų būtų priskirti skirtingose kategorijose. Artimiausias šiems reikalavimams Epinions.com duomenų rinkinys naudotas šiame tyrime netenkina šių dviejų reikalavimų - tai yra viena priežasčių, kliudžusių atlikti išsamesnį tyrimą su realiais duomenimis. Dėl šios priežasties, nemaža tyrimo dalis skirta duomenų rinkinio generavimui.

Kita problema susijusi su RS vertinimu. Negalima vienareikšmiškai apibrėžti, kokia RS yra gera. Egzistuoja nemažai kriterijų, pagal kuriuos galime vertinti RS - tiek tikslumas ir kriterijai, kuriuos jis apima (vidutinė absoliuti klaida, vidutinė kvadratinė klaida, normalizuoti šių matų atitikmenys), tiek ir tam tikrų savybių tenkinimas (naujoviškumas, įžvalgumas, tikslumas, atsparumas atakoms, padengimas), tačiau RS kūrėjai turi apsispręsti, kurie kriterijai yra svarbesni, o kurie mažiau svarbūs. Kitaip sakant, reikia atsakyti į tokius klausimus kaip: ar geriau sistema generuotų tikslias rekomendacijas net jeigu naudotojas jau žino apie visus elementus iš ankščiau ar jau vėčiau kartais suklysta, bet dažnai pasiūlo kažką naujo? Priimant sprendimą būtina atsižvelgti į dalykinę sritį. Vis dėlto, parinkti tinkamus reikalavimus yra didelis iššūkis analitikams, nes reikia atsižvelgti ne tik sistemos tikslumą, bet ir žmonių reakcijas į rekomendacijas. Kadangi šio tyrimo tikslas - iširti metodus, siekiančius padėti sudaryti rekomendacijas mažai duomenų turintiems naudotojams,

buvo koncentruotasi ties dviem RS vertinimo aspektais - tikslumu ir padengimu.

Trečia problema - technologinė. Darbas su dideliais grafais reikalauja technologijų optimizuo-  
tų tokiems duomenims. Dėl šios priežasties tyrimas buvo atliktas su nedidelės apimties imtimi.  
Algoritmus realizuojantis kodas buvo parašytas .NET aplinkoje C# ir F# kalbomis, duomenys sau-  
gomi ir kai kurios grafų operacijos (pavyzdžiui, trumpiausio kelio radimas) atliekamos NoSql neo4j  
grafų duomenų bazėje. Tinkamų technologijų parinkimas ir architektūros sudarymas šiame tyrime  
nagrinėtiems uždaviniams spręsti - potenciali tolimesnė šio tyrimo dalis.

### 3. Išvados

Didžioji dauguma tyrimų apie RS, BF ir pasitikėjimu pagrįstas RS buvo atlikta vienmatėje aplinkoje - daroma prielaida, kad RS dalykinė sritis yra vienalytė ir naudotojų tarpusavio panašumas arba pasitikėjimas yra vienalytis. Šiame darbe siūloma RS padalinti pagal pasitikėjimo sritis ir taip pakeisti pasitikėjimo įvertį iš skaliaro į vektorių. Toks aplinkos transformavimas įgalina naudoti du darbe pasiūlytus metodus.

Sričių panašumo metodas leidžia įvertinti pasitikėjimą nežinomoje srityje, kai yra žinomas pasitikėjimas kitoje ir šių sričių tarpusavio panašumo įvertis. Taikant šį metodą atsiranda galimybė pasiūlyti rekomendaciją ne tik to, ką palankiai įvertino naudotojai, kuriais pasitikime tam tikroje srityje, bet ir tai ką jie gerai įvertino ir kitoje srityje. Tai yra ypač aktualu esant šaltam startui - sistema apie naudotoją žino nedaug, nes metodo taikymas praplečia galimų rekomendacijų aibę.

Pasitikėjimo apskaičiavimas taikant tiesinę regresiją - kitas metodas leidžiantis įvertinti nežinomą pasitikėjimo įvertį vienoje srityje, kai yra žinomi pasitikėjimo įverčiai kitose. Šis metodas naudoja prielaidą, kad pasitikėjimas yra abipusis, tai yra, neturi krypties (ši prielaida kai kurioms dalykinėms sritims yra teisinga). Taikant tiesinę regresiją apskaičiuojamas naudotojo pasitikėjimas naudotoju, susiduriančiu su šalto starto problema ir tada jam priskiriamas pasitikėjimo įvertis.

Darbe pasiūlytas dar vienas metodas, kuris nenaudoja kelių pasitikėjimo sričių apibrėžimo. Bendrų kaimynų metodas taikomas, kai norime įvertinti vieno naudotojo pasitikėjimą kitu, tačiau jie neturi tiesioginio ryšio, o pasitikėjimo tinkle nėra jokių pasitikėjimo įverčių, tai yra, viskas, ką žinome apie konkretų naudotoją - jo ryšiai. Metodo esmė - panaudoti dviejų naudotojų bendrų ir savo ryšių skaičiaus santykį prognozuojant pasitikėjimą, o tada, remiantis prognozuojamu pasitikėjimu, įvertinti reitingų prognozę taikant bendradarbiavimo filtravimo metodą.

Atlikus tyrimą paaiškėjo, kad pasitikėjimo prognozės tiksliausios, kai yra didelės bendrų ir naudotojų ryšių skaičiaus santykio reikšmės. Galutiniai rezultatai pagal *MAE* ir *RMSE* kriterijus labai panašūs į tuos, kuriuos gauname pritaikę bendradarbiavimo filtravimo algoritmą. *MAUE* kriterijaus reikšmė kiek didesnė, o tai reiškia, kad metodas prasčiau veikia naudotojams, turintiems mažiau reitingų. Sudarant pasitikėjimų prognozę reikia sudaryti imtį iš naudotojų, turinčių pakankamai daug ryšių - tada tiek pasitikėjimo prognozė, tiek gautinės rekomendacijos būna tikslesnės. Bendrų kaimynų metodas turėtų būti naudojamas tais šalto starto atvejais, kai apie naudotoją, kuriam norime kažką rekomenduoti, yra žinomi tik jo ryšiai su kitais naudotojais. Taip pat prasminga nustatyti slenkstį, nurodantį naudotojo ryšių skaičių, nes kuo daugiau ryšių turi naudotojas, tuo prognozės tikslumas didesnis.

Pasiūlyti metodai sprendžia ne tik aptartą šalto starto problemą, bet ir kitą kertinę bėdą, su kurią

susiduria visos RS - duomenų retumo ir nepakankamumo. Jų taikymas leidžia panaudoti turimus duomenis situacijose, kai įprasti tradiciniai metodai negali veikti.

## Literatūros sąrašas

- [1] Pasquale Lops, Marco de Gemmis, Giovanni Semarero *Content-based Recommender Systems: State of the Art and Trends* Recommender Systems Handbook, 73-100, 2010.
- [2] Christian Desrosiers, George Karypis *A Comprehensive Survey of Neighborhood-based Recommendation Methods* Recommender Systems Handbook, 101-140, 2010.
- [3] Guy Shani, Asela Gunawardana *Evaluating recommender systems* Recommender Systems Handbook, 257-298, 2010.
- [4] Robin Burke, Michael P. O'Mahony, Neil J. Hurley *Robust Collaborative Recommendation Systems: State of the Art and Trends* Recommender Systems Handbook, 805-836, 2010.
- [5] Patricia Victor, Martine De Cock, Chris Cornelis *Trust and Recommendations Systems: State of the Art and Trends* Recommender Systems Handbook, 645-676, 2010.
- [6] Paolo Massa, Paolo Avesani *Trust-aware recommender systems* Proceedings of the 2007 ACM conference on Recommender systems (2007) 17-24
- [7] Hyung Jun Ahn *A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem* Information Sciences Vol 178 (2008) 37-51
- [8] Jon Herlocker, Joseph A. Konstan, John Riedl *An Empirical Analysis of Design Choices in Neighborhood-Based Collaborative Filtering Algorithms* Information Retrieval 5 178 (2002) 287-310
- [9] Michael D. Ekstrand, John T. Riedl, Joseph A. Konstan *Collaborative filtering recommender systems* Foundation and trends in Human-Computer Interaction Vol. 4, No. 2 (2010) 81-173
- [10] Jennifer Ann Golbeck *Computing and applying trust in web-based social networks* Dissertation
- [11] Paolo Avesani, Paolo Massa, Roberto Tiella *A Trust-enhanced Recommender System application: Moleskiing* Proceedings of the 2005 ACM symposium on Applied computing (2005) 1589-1593
- [12] John O'Donovan, Barry Smith *Trust in Recommender Systems* Proceedings of the 10th international conference on Intelligent user interfaces (2005) 167-174

- [13] Alan Said, Brijnesh J. Jain, Sahin Albayrak *Analyzing Weighting Schemes in Collaborative Filtering: Cold Start, Post cold Start and Power Users* Proceedings of the 27th Annual ACM Symposium on Applied Computing (2012) 2035-2040
- [14] Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, John Riedl *An Algorithmic Framework for Performing Collaborative Filtering* Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval (1999) 230-237
- [15] Sergio Mateo Maria *Collaborative Filtering in social Networks* (2010)
- [16] David Goldberg, David Nichols, Brian M. Oki, Douglas Terry *Using collaborative filtering to weave an information tapestry* Communications of the ACM - Special issue on information filtering CACM Homepage archive Volume 35 Issue 12 (1992) 61-70
- [17] Cai-Nikolas Ziegler, Georg Lausen *Propagation Models for Trust and Distrust in Social Networks* Information Systems Frontiers December 2005, Volume 7, Issue 4 Volume 35 Issue 12 (2005) 337-358
- [18] Audin Josang, Stephen Marsh, Simon Pope *Exploring Different Types of Trust Propagation* Proceedings of the 4th international conference on Trust Management (2006) 179-192
- [19] Sinha, Rashmi R., and Kirsten Swearingen. *Comparing Recommendations Made by Online Systems and Friends* DELOS workshop: personalisation and recommender systems in digital libraries. Vol. 1. 2001.