

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Analýza emailové komunikace

Analysis of Email Communication

2018

Veronika Uhrová

Zadání diplomové práce

Student:

Bc. Veronika Uhrová

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2612T025 Informatika a výpočetní technika

Téma:

Analýza emailové komunikace

Analysis of Email Communication

Jazyk vypracování:

čeština

Zásady pro vypracování:

Cílem práce je návrh a implementace systému na analýzu emailové komunikace a vizualizaci výstupů. Systém bude pracovat s reálnými daty a budou navrženy, popsány a vyhodnoceny experimenty s těmito daty. Pro implementaci je doporučen jazyk C#.

1. Rešerše obdobných řešení a analytických přístupů.
2. Návrh a implementace metody na získávání emailových zpráv z vybraného zdroje.
3. Výběr a implementace metod strojového učení a analýzy sítí vhodných pro analýzu emailové komunikace.
4. Návrh a implementace uživatelského rozhraní na analýzu emailové komunikace a vizualizaci analytických výstupů.
5. Dokumentovaná implementace systému.

Seznam doporučené odborné literatury:

- [1] S. Zehnalova, Z. Horak, M. Kudělka. Email Conversation Network Analysis: Work Groups and Teams in Organizations. ASONAM 2015.
[2] Tang, G., Pei, J., Luk, W. S. (2014). Email mining: tasks, common techniques, and tools. Knowledge and Information Systems, 41(1), 1-31.

Další podle pokynů vedoucího práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Mgr. Miloš Kudělka, Ph.D.**

Datum zadání: 01.09.2017

Datum odevzdání: 30.04.2018

doc. Ing. Jan Platoš, Ph.D.
vedoucí katedry



prof. Ing. Pavel Brandstetter, CSc.
děkan fakulty

Prehlasujem, že som túto prácu vypracovala samostatne. Uviedla som všetky literárne pramene a publikácie, z ktorých som čerpala.

V Ostrave, 27.4.2018

.....

Moje podakovanie patrí predovšetkým doc. Milošovi Kudělkovi, Ph.D. za odborné konzultácie a vedenie mojej diplomovej práce.

Abstrakt

Práca študuje aktuálne metódy pre analýzu emailov a detekciu sociálnych rolí v emailových dátach. Nasleduje zoznamenie sa s emailom a jeho popularitou v súčasnosti. Taktiež práca uvádza základné teoretické pojmy a teoretický náhľad na reprezentáciu siete. Uvádzajú sa tu aj základy analýzy sociálnych sietí a detektie komunit. Ďalej sa tu píše o framework pre detekciu štrukturálnych rolí a ich identifikácie. Ďalšou popísanou metódou pre analýzu sietí je identifikácia *brokerage* rolí. Na základe týchto poznatkov je vytvorená aplikácia pre analýzu a vizualizáciu analytických výstupov. Na záver sú uvedené prevedené experimenty.

Klúčové slová: email, sociálna sieť, sociálna rola, ego sieť, vizualizácia, brokerage

Abstract

This paper studies current methods for analysing emails and detecting social roles in email data. This is followed by getting acquainted with the email and its popularity nowadays. Also, this thesis presents basic theoretical concepts and a theoretical overview of network representation. Here are also the basics of social networking and community detection. There is also written a framework for structural social roles detection and their identification. Another method for social network analysis is identification of brokerage roles. Based on this knowledge, an application is developed to analyze and visualize analytical outputs. Finally, experiments on the findings of the emailed data are presented.

Key Words: email, social network, social role, ego network, visualization, brokerage

Obsah

Zoznam použitých skratiek a symbolov	10
Zoznam obrázkov	11
Zoznam tabuliek	13
1 Úvod	14
1.1 Motivácia	14
1.2 Vízia	14
1.3 Štruktúra práce	15
2 Súvisiace práce	16
3 Emailová komunikácia	18
3.1 Stručná história emailu	18
3.2 Štruktúra emailu	18
3.3 Emaily v súčasnosti	19
4 Definície a klasifikácie	20
4.1 Graf	20
4.2 Súvislost grafu	21
4.3 Úplný graf	21
4.4 Stupeň vrcholu	21
4.5 Cesta	22
4.6 Uzavretá cesta	22
4.7 Komponenta grafu	22
4.8 Metriky	22
4.8.1 Closeness centrality (Centralita blízkosti)	22
4.8.2 Betweenness centrality (Centralita medziľahosti)	22
4.8.3 Modularita	23
5 Sociálna sieť	24
5.1 História sociálnych sietí	24
5.2 Analýza sociálnych sietí	24
5.3 Komunity v sociálnych sietach	25
5.3.1 H2: Predpoklad súvislosti a hustoty	25
5.3.2 Maximálne kliky	26
5.4 Silné a slabé komunity	26
5.5 Detekcia komunít	27

5.5.1	Louvainov algoritmus pre detekciu komunít	27
5.6	Ego siet	29
5.6.1	Konštrukcia ego siete	29
6	Metódy analýzy sociálnych sietí	30
6.1	SSRM - Framework pre detekciu štrukturálnych rolí v sociálnych sietach	30
6.1.1	Rola v kontexte SSRM	30
6.1.2	Roly definované v SSRM	30
6.1.2.1	Leader	31
6.1.2.2	Outermost	31
6.1.2.3	Mediator	31
6.1.2.4	Outsider	31
6.2	Identifikácia štrukturálnych sociálnych rolí	31
6.2.0.1	Outsider	32
6.2.1	Leader	32
6.2.1.1	Closeness centrality (Centralita blízkosti)	32
6.2.2	Outermost	32
6.2.3	Mediator	32
6.2.3.1	LBeweeness	32
6.2.3.2	CBetweenness	33
6.2.3.3	Normalizovaná verzia CBetweenness	33
6.2.3.4	Skóre rozmanitosti	34
6.3	Brokerage roly	34
6.3.1	Liaison	36
6.3.2	Itinerant	36
6.3.3	Coordinator	36
6.3.4	Gatekeeper	37
6.3.5	Representative	37
6.3.6	Identifikácia brokerage rolí	37
6.3.7	Popis metódy pre identifikáciu brokerage rolí	38
6.4	Analýza ega	39
6.4.1	Veľkosť ego siete	39
6.4.2	Kompozícia ego siete	39
6.4.3	Štruktúra ego siete	40
6.4.3.1	Efektívna veľkosť	40
7	Aplikácia	42
7.1	Špecifikácia	42
7.1.1	Funkčné požiadavky	42
7.2	Návrh	43

7.2.1	Návrhové vzory	44
7.3	Dôležité rozhodnutia	45
7.3.1	Dostupnosť dát	45
7.3.2	Webová vs. desktopová aplikácia	45
7.4	Použité knižnice	46
7.5	Import dát	47
7.6	Implementácia	48
7.6.1	Metóda pre získanie emailových dát	48
7.6.2	Konštrukcia siete	49
7.6.3	Triedy pre graf, vrcholy a hrany	49
8	Experimenty	50
8.1	Analýza emailovej komunikácie tímu	50
8.1.1	Príprava a import dát	50
8.1.2	Vizualizácia datasetu	51
8.1.3	Detekcia komunít	52
8.1.3.1	Zmeny komunít v čase	54
8.1.4	Ego siet	55
8.1.5	Analýza rolí	55
8.1.5.1	SSRM	55
8.1.5.2	Brokerage	56
8.2	Analýza jednotlivca	57
8.2.1	Príprava a import dát	57
8.2.2	Informácie o datasete	58
8.2.3	Detekcia komunít	59
8.2.3.1	Zmeny komunít v čase	60
8.2.4	Ego siet	61
8.2.5	Analýza rolí	61
8.2.5.1	SSRM	61
8.2.5.2	Brokerage	62
9	Záver	64
9.1	Možnosti rozšírenia a zdokonalenia práce	64
9.1.1	Možné rozšírenia aplikácie	64
Literatúra		65
10 Prílohy		67

Zoznam použitých skratiek a symbolov

MUA	– Mail User Agent
MTA	– Mail Transfer Agent
IMAP	– Internet Message Access Protocol
XML	– eXtensible Markup Language
SSRM	– Structural social role mining framework
SNA	– Social network analysis

Zoznam obrázkov

1	Akú formu komunikácie preferujete na formálnu komunikáciu?	19
2	Neorientovaný graf	20
3	Orientovaný graf	20
4	Súvislý (1) a nesúvislý graf (2)	21
5	Úplný graf	21
6	Graf v tvare hviezdy	23
7	Komunity	25
8	Vizualizácia krokov Louvainovho algoritmu.	28
9	Príklad ego siete.	29
10	Príklad brokerage procesu	35
11	Liaison brokerage	36
12	Itinerant brokerage	36
13	Coordinator brokerage	36
14	Gatekeeper brokerage	37
15	Representative brokerage	37
16	Identifikácie <i>brokerage</i> rolí [1]	38
17	Velkosť ega - stupeň uzla: 6	39
18	Málo štrukturálnych dier vs. veľa štrukturálnych dier.	40
19	Príklad výpočtu redundancie	41
20	UseCase Diagram	43
21	Diagram komponent znázorňujúci jednotlivé komponenty architektúry aplikácie .	44
22	Triedny diagram - Repository pattern	44
23	Model-View-Controller	45
24	Jednoduchá sieť vytvorená s použitím knižnice vis.js	46
25	Príklad použitia knižnice vis.js	47
26	Doménový model	48
27	Príklad konfigurácie emailu pre získanie emailov	49
28	Základné informácie o tímovej sieti.	51
29	Najviac používané emailové domény.	51
30	Vizualizácia siete.	52
31	Vizualizácia komunít v tímovej sieti za celkový čas	53
32	Rozloženie komunít v tímovej sieti za celkový čas	53
33	Rozloženie komunít za prvý časový úsek	54
34	Rozloženie komunít za druhý časový úsek	54
35	Rozloženie komunít za tretí časový úsek	55
36	Počet detekovaných štrukturálnych rôl	56
37	Desať aktérov s najväčším <i>brokerage</i> skórom	56

38	Desať aktérov s najväčším <i>brokerage</i> skórom - graf	57
39	Analýza jednotlivca - základná vizualizácia	58
40	Analýza jednotlivca - základné štatistiky	59
41	Analýza jednotlivca - vizualizácia komunít	59
42	Analýza jednotlivca - Vizualizácia komunít v prvom časovom intervale	60
43	Analýza jednotlivca - vizualizácia komunít v druhom časovom intervale	60
44	Analýza jednotlivca - detail detekovaných SSRM rolí	62
45	Desať aktérov s najväčším <i>brokerage</i> skórom	62
46	Desať aktérov s najväčším <i>brokerage</i> skórom - graf	63

Zoznam tabuliek

1	Základné informácie o datasete	50
2	Informácie o členoch tímu	52
3	Informácie o vytvorennej ego sieti	55
4	Informácie o vytvorennej ego sieti	61

1 Úvod

V stručnom úvode je popísaná motivácia, ktorá viedla k vypracovaniu tejto diplomovej práce a vízia toho, čo sa malo dosiahnuť a hrubá štruktúra vypracovaného textu.

1.1 Motivácia

S cieľom uľahčiť používanie emailov a prebádať podnikateľský potenciál emailov, analýza emailov dosiahla pozoruhodný pokrok nielen v oblasti výskumu, ale aj v praxi. Emaily možno považovať za zmiešanú štruktúru obsahujúcu údaje o ľuďoch zo sociálnych alebo aj organizačných aspektov.

Obsah emailu ako textové a netextové dát

Emaily sú písané viac stručne ako väčšina ostatných dokumentov, často obsahujú hovorové výrazy a abreviácie, ktoré sa nenachádzajú v bežných slovníkoch, preto štandardné techniky analýzy textov pri práci s emailovými dátami nemusia byť efektívne.

Emaily tiež obsahujú bohatšie typy dát, ako napríklad URL linky, HTML tagy alebo obrázky. Niektoré štúdie jednoducho zjednodušia tieto netextové dátové vstupy v štádiu predpripravovania dát - vymažu ich a ďalej pracujú len s textovými dátami. Tieto netextové dátá však môžu byť užitočné v iných oblastiach, ako napríklad detekcia spamu.

Emaily reprezentujúce ľudské sociálne organizačné vzťahy

Emailová aktivita sama o sebe reprezentuje bohaté ľudské sociálne a organizačné vzťahy, ktoré spájajú ľudí do komunít a komplexných systémov. Porozumenie organizačných štruktúr alebo vzťahov naprieč ľuďmi v organizácii môže byť veľmi užitočné aj v reálnom živote. Hlavné problémy, ktoré sú investigované v analýze emailov sú detekcia spamu, kategorizácia emailov, analýza kontaktov, analýza vlastností emailových sietí a vizualizácia emailov.

1.2 Vízia

Cieľom práce je oboznámiť čitateľa s oblasťou sociálnych sietí a špeciálne s tému analýzy emailových dát a tieto znalosti demonštrovať nad reálnymi emailovými dátami. Pre uskutočnenie tohto cieľa je potrebné naštudovať informácie z oblasti analýzy emailov, reprezentácie emailu v sieti a vizualizácie sociálnych sietí vrátane aktuálnych metód publikovaných v článkoch. K tomu sa viaže tiež prieskum reprezentácie a konštrukcie emailu ako prvku sociálnej siete.

Ďalej boli vybrané metódy detekcie rolí v sociálnej sieti a navrhnutá aplikácia, ktorá umožňuje analyzovať a vizualizovať analytické výsledky. V tejto aplikácii s jednoduchým a použiteľným užívateľským rozhraním sú implementované vybrané metódy analýzy a je navrhnutá prehľadná vizualizácia vzťahov. Nakoniec je vytvorená analýza tímu podľa emailových dát a porovnanie dvoch prvkoch siete a výsledky experimentov sú zrozumiteľne prezentované.

1.3 Štruktúra práce

V prvej kapitole je uvedený prieskum o aktuálnych vedeckých článkoch, ktoré sa zaobrajú analýzou emailov a reprezentáciou emailu v sociálnych sieťach. Ďalej sa čitateľ zoznámi s emailom ako komunikačným prostriedkom a dozvie sa, ako sú na tom emaily s popularitou aktuálne. Potom je uvedený stručný prehľad teórie grafov a definícií určitých pojmov, ktorý je nevyhnutný k porozumeniu ďalších kapitol. V ďalšej kapitole píšem o sociálnych sieťach, ich histórii a analýze sociálnych sietí, komunitnej štruktúre sociálnych sietí a ego sietach. Neskôr prechádzam k popisu a reprezentácii frameworku pre detekciu štrukturálnych rolí, popisujem sociálne roly definované v rámci tohto frameworku a následne v ďalšej kapitole referujem pomocou akých metód sa sociálne roly v rámci tohto frameworku identifikujú. Ďalej popisujem ďalšiu metódu pre identifikáciu rolí zo sociálnych sietí - *brokerage*. Na základe všetkých poznatkov práce je navrhnutá aplikácia vhodná k sledovaniu výsledkov navrhnutých metód pre analýzu emailových sietí. Ešte pred záverom sú uvedené výsledky prevedených experimentov týkajúcich sa poznatkami skúmanej sociálnej siete.

2 Súvisiace práce

Pre odhalovanie vzťahov medzi ľuďmi, skupinami a organizáciami z emalových sietí boli aplikované mnohé techniky a modely analýzy sociálnych sietí. Mnoho štúdií použilo maily spoločnosti *Enron* kvôli nedostatku dostupných veľkých súborov.

Napríklad Diesner, Carley a Frantz v [2] zkonštruovali z mailovej komunikácie spoločnosti Enron orientovaný graf zo vzťahu odosielateľ-príjemca, kde hrany boli vážené frekvenciou mailov, ktoré si medzi sebou poslali v čase. Potom aplikovali techniky analýzy sociálnych sietí. V práci popísali, ako vylepšili originálnu sadu a súčasné zistenia ich investigáciou vďaka analýze sociálnych sietí. Skúmajú dynamiku, štruktúru a vlastnosti organizačnej komunikačnej siete ako aj charakteristiky a vzory komunikačného správania zamestnancov z rôznych organizačných levelov. Zistili, že počas obdobia krízy sa komunikácia medzi zamestnancami stala viac rôznorodejšia v súvislosti so zavedenými kontaktami a formálnymi rolami. Taktiež počas obdobia kríz, predtým nekomunikujúci zamestnanci sa začali zapájať do vzájomného rozhovoru, takže interpersonálna komunikácia bola intenzívnejšia a sieť sa tým rozširovala. Tieto zistenia poskytli cenný pohľad do organizačnej krízy reálneho sveta, čo môže byť ďalej využité pre validáciu alebo tvorbu teórií a dynamických modelov organizačných kríz a tým to vedie k lepšiemu porozumeniu základných príčin organizačných kríz v organizáciách.

Xiaoyan Fu v [3] prezentoval rôzne metódy pre vizualizáciu emailových sietí. Vizualizácia objavuje komunikačné vzory medzi rôznymi skupinami, zobrazuje centrálnu analýzu s dôrazom na významné uzly. V práci zkonštruovali 2D vizualizáciu temporálnej emailovej siete, ktorá analyzuje vývoj emailových vzťahov, ktoré sa menia v priebehu času a zobrazenie prostredia pre nájdenie sociálnych kruhov odvodených od siete. Každá metóda bola vyhodnotená s rôznymi datasetmi od výskumnej organizácie. Taktiež rozšírili ich metódu pre vizuálnu analýzu siete emailových vírusov.

Ďalej Chapanond, Krishnamoorthy, Yener v [4] použil sietové metriky a spektrálnu analýzu k analýze či už orientovaného alebo neorientovaného grafu emailov, ktorý skonštruoval zmenou prahovej hodnoty (napr. počtom vymenených emailov medzi užívateľmi). Ich výskum je postavený na vytvorení emailového grafu a štúdiu jeho vlatnosťí či už pomocou teórie grafov alebo technikami spektrálnej analýzy. Grafová teoretická analýza zahŕňa výpočet niekolkých grafových metrík, ako napríklad rozdelenie podľa stupňov, priemerný pomer vzdialenosťí, zhlukovací koeficient alebo kompaktnosť emailového grafu. Hodnoty metrík v dátovej sade emailov spoločnosti Enron porovnali aj s inými emailovými dátami.

Jednou z univerzálniejsích prác je aj práca autorov Guanting Tang, Jian Pei, and Wo-Shun Luk [5]. Je to stručný prehľad hlavných výskumných snáh o analýzu mailov a popis metód, ktoré sa pri tejto analýze používajú. Nie len čo sa týka analytických alebo implemetnačných úloh, ale aj nástrojov, ktoré nám pri analýze vedia pomôcť. Aby zdôraznili rozdiely medzi analýzou mailov a bežnou analýzou textu, organizujú prieskum do piatich ľažších úloh a to: detekcia nevyžiadanej pošty, kategorizácia emailov, analýza kontaktov, analýza vlastností emailovej siete

a vizualizácia emailov. Tieto úlohy sú vlastne začlenené do rôznych spôsobov používania emailov. Systemaicky preskúmavajú bežne používané techniky a tiež budujú diskusiu o dostupných softwarových nástrojoch.

Na rozdiel od ostatných prác, Afra Abnar, Mansoureh Takaffoli, Reihaneh Rabbany, Osmar R. Zaiane [6] definovali vlastnú metodiku pre analýzu sociálnej siete a definovali *Structural social role mining framework*, ktorý je navrhnutý pre identifikáciu štrukturálnych rolí, pre identifikáciu zmien v sieti a analýzu dopadu zmien na sieť. Definujú základné sociálne roly v sieti a navrhujú metodológie pre ich identifikáciu. Pre identifikáciu týchto rolí využívajú klasické prostriedky analýzy sociálnych sietí a tiež navrhujú nové metriky zahrňujúc napríklad Betweenness centrality založenú na komunitách. Z tejto práce som vychádzala pri pomenovaní rolí zo siete a implementovala techniky pre ich identifikáciu.

Ďalšou prácou, ktorou som sa inšpirovala bola práca autorov Kudělka, Horák, Zehnalová [7], ktorá prezentuje analytický nástroj, ktorý bol vytvorený pre analýzu hlbších vzťahov v emailových dátach. Tieto vzťahy zahrňujú vzťahy založené na interakcii viacerých užívateľov v tíme. Analytické metódy popísané v práci sú založené na dvoch faktoroch. Prvým faktorom je kontext, čo je skupina viacerých užívateľov v kombinácii so slovami použitými v komunikácii. Druhým faktorom je časový interval, v ktorom bola začatá komunikácia. Práca prezentuje metódy pre väženie komunikácií, užívateľov a vzťahov, ako aj metód pre hľadanie komunít asociovaných so špecifickým kontextom.

3 Emailová komunikácia

3.1 Stručná história emailu

Za počiatky emailovej komunikácie možno považovať priližne rok 1965, kedy bola správa prenášaná medzi sálovými počítačmi pracujúcich v režime zdieľania času na univerzite *Massachusetts Institute of Technology*.

Od tejto doby preša emailová komunikácia značným vývojom. Emaily, tak ako ich poznáme dnes, sú definované štandardom špecifikácie RFC2822 a sú prenášané pomocou komunikačných protokolov.

3.2 Štruktúra emailu

Každý email sa skladá z dvoch častí - z tzv. hlavičky (*header*) a tela emailu (*body*).

Hlavička emailu je generovaná automaticky pri vytvorení emailu a sú do nej postupne vkladané informácie zo serverov, cez ktoré správa prechádza (tzv. MTA). Pre bežných užívateľov sú z hlavičky najdôležitejšie tieto údaje: predmet správy, čas odoslania, emailová adresa odosielateľa a prijímateľa. Ostatné údaje emailoví klienti (označovaní tiež ako MUA¹) väčšinou nezobrazujú.

Pri vytváraní emailu emailovým klientom sú väčšinou do hlavičky vložené tieto záhlavia:

- **Date** - aktuálny čas počítača, ktorý vložil záhlavie
- **From** - adresa odosielateľa
- **Cc** - špecifikuje ďalších adresátov
- **Bcc** - umožňuje rozosielanie správy medzi viacerých adresátov
- **Priority** - priorita emailu, interpretácia sa lísi vzhľadom k MUA
- **Reply-To** - špecifikuje adresu, na ktorú je zaslaná prípadná odpoved
- **Subjekt** - predmet správy daný užívateľom
- **To** - udáva adresu príjemcu správy
- **Message-ID** - unikátny identifikátor, ktorý je priradený MTA

Telo emailu obsahuje samotné dátá určené pre adresáta.

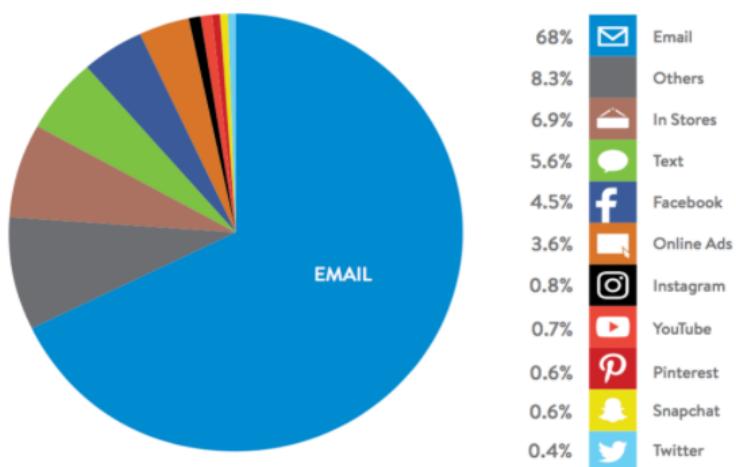
¹MUA - Mail User Agent, program, ktorý používa užívateľ na rozosielanie a prijímanie emailov (napr. Outlook), tento program komunikuje s MTA (Mail Transfer Agent), ktorý sa stará o prenos emailov v prostredí verejnej siete Internet.

3.3 Emaily v súčasnosti

Emaily teda existujú už niečo cez 50 rokov, ich popularita je však stále veľká vďaka ich efektivite, extrémne nízkym nákladom a kompatibilite s množstvom typov zariadení. Ako jedna z najrozšírenejších typov komunikácie v dnešnej dobe, emaily sú široko rozšírené v každodennom živote. Napríklad, spolupracovníci diskutujú prácu cez emails, priatelia zdielajú sociálne aktivity a skúsenosti aj cez emails alebo veľké spoločnosti distribuujú reklamy práve pomocou emailov.

Aj keď by mnohí tvrdili, že éra emailov už je dávno preč a sú stále viac nahradzane novými sociálnymi sieťami, nové výskumy ukazujú opak. Napríklad výskum z roku 2016 od spoločnosti Bluecore [8] ukazuje, že email je stále populárny aj u mladších generácií, hlavne na formálnej komunikácii.

V tomto výskume boli spotrebiteľia pýtaní, akú formu komunikácie preferujú pri komunikácii so značkami (internetovými obchodmi, na firemnú komunikáciu a celkovo formálnej komunikácii). Prevažná časť opýtaných si vybrała email (68%).



Obr. 1: Akú formu komunikácie preferujete na formálnej komunikácii?

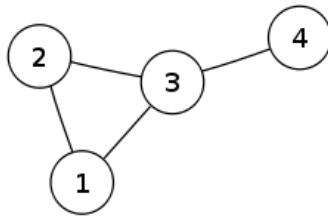
4 Definície a klasifikácie

V tejto kapitole popisujem všetky teoretické pojmy a metódy, ktoré v tejto práci spomínam a používam. V tejto kapitole budem používať matematické názvy podľa kontextu, v ktorom sa budem nachádzať.

4.1 Graf

- **Neorientovaný graf**

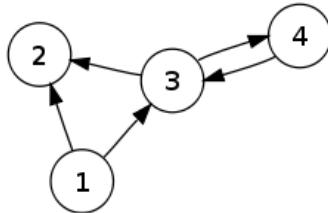
Neorientovaným grafov rozumieme usporiadanie dvojicu $G = (V, E)$, kde V je neprázdna množina *vrcholov* a E je neprázdná množina *hrán* - množina (niektorých) dvojprvkových podmnožín množiny V .



Obr. 2: Neorientovaný graf

- **Orientovaný graf**

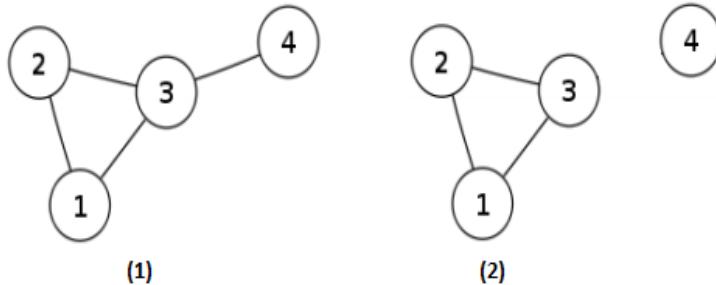
Orientovaným grafov rozumieme usporiadanie dvojicu $G = (V, E)$, kde V je množina *vrcholov* a množina orientovaných *hrán* je $E \subseteq V \times V$.



Obr. 3: Orientovaný graf

4.2 Súvislosť grafu

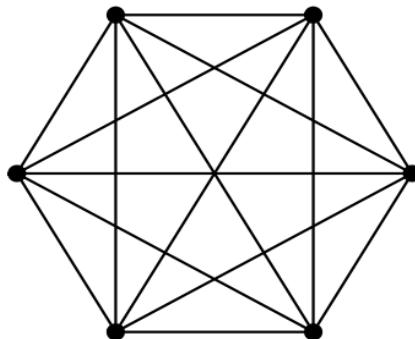
Hovoríme, že vrchol v je dosiahnuteľný z vrcholu u , ak v grafe existuje sled z vrcholu u do vrcholu v . Graf nazveme *súvislý*, ak pre každé dva vrcholy u, v je vrchol v dosiahnuteľný z vrcholu u . V opačnom prípade je graf *nesúvislý*.



Obr. 4: Súvislý (1) a nesúvislý graf (2)

4.3 Úplný graf

Úplný graf na n vrcholoch je neorientovaný graf, ktorý má hranu medzi každými dvoma vrcholmi. Počet jeho hrán je $m = n(n - 1)/2$.



Obr. 5: Úplný graf

4.4 Stupeň vrcholu

Stupeň vrcholu je počet vrcholov spojených s týmto vrcholom hranou, inými slovami: počet jeho susedov. V orientovanom grafe sa ešte rozlišuje vstupný a výstupný stupeň vrcholu podľa toho, koľko hrán z vrcholu vychádza alebo do neho vchádza. Stupeň vrcholu u je $\deg(u) = |\{e \in E | u \in e\}|$

4.5 Cesta

Cesta je postupnosť vrcholov v grafe taká, že medzi každými dvoma vrcholmi cesty je hrana a vrcholy sa neopakujú. V orientovaných grafoch sa ešte rozlišuje smer cesty, pričom orientácia hrán je stále rovnaká. Dĺžka cesty je počet hrán, ktoré obsahuje.

4.6 Uzavretá cesta

Uzavrená cesta, kružnica v neorientovanom a cyklus v orientovanom grafe, je cesta, ktorá začína a končí v rovnakom uzle.

4.7 Komponenta grafu

Komponenta grafu je súvislá časť grafu a medzi vrcholmi z rôznych komponent neexistuje žiadna hrana.

4.8 Metriky

V tejto časti popisujem metriky, ktoré v rámci identifikácie rolí v sieti používam. Ďalšie informácie o metrikách, ich praktickom využití a ich ďalších variantách sú zhrnuté v kapitole 6.2.

4.8.1 Closeness centrality (Centralita blízkosti)

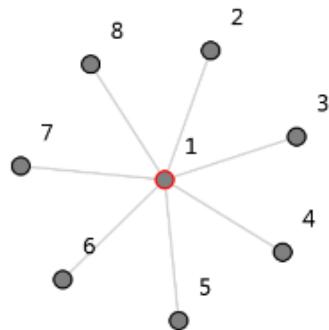
Táto centralita meria dôležitosť vrcholu grafu podľa priemernej hodnoty vzdialenosť od všetkých ostatných vrcholov v sieti. Aby dôležité vcholy mali vyššie číslo, je táto centralita počítaná ako inverzná hodnota tohto priemeru. Vrchol dôležitý podľa tejto metriky môže mať dobrý prístup k informáciám o ostatných vrcholoch alebo naopak môže ostatné vrcholy rýchlosťou ovplyvňovať.

Priemernú vzdialenosť vrcholu x_i od ostatných vcholov možno formálne zapísat ako $l_i = \frac{1}{n} \sum_j d_{ij}$, kde n je počet vrcholov v grafe a d_{ij} je najkratšia cesta medzi vrcholmi x_i a x_j . Centralita je potom $C_i = \frac{1}{l_i}$.

4.8.2 Betweenness centrality (Centralita medziľahlosti)

Táto centralita sa odlišuje od ostatných uvedených. Jej hodnota pre vrchol je počet najkratších ciest medzi každými dvoma vrcholmi v grafe, na ktorých hodnotený vrchol leží. Pokiaľ medzi vrcholmi v sieti tečú nejaké informácie alebo sa posielajú správy, hodnota tejto metriky vyjadzuje, aké množstvo informácií cez daný vrchol prejde. Táto centralita je tiež názorný príklad toho, že každá metrika počíta dôležitosť vrcholu úplne inak. Vrchol s vysokou centralitou medziľahlosti môže mať malý stupeň a nemusí ležať blízko ostatných vrcholov, stačí, keď cez neho prechádza veľa najkratších ciest. To môže nastať, pokiaľ vrchol je most medzi dvoma alebo viacerými komponentami v grafe, v extrémnom prípade pokiaľ je v strede grafu v tvare hviezdy (viď obrázok).

Betweeness vrcholu x_i možno spočítať ako $B_i = \sum_{st} \frac{n_{st}^i}{g_{st}}$, kde g_{st} je počet všetkých najkratších ciest medzi vrcholmi x_i a x_j a n_{st}^i je počet najkratších ciest, ktoré naviac vedú cez vrchol x_i .



Obr. 6: Graf v tvare hviezdy

4.8.3 Modularita

Modularita je metrika, ktorá udáva rozdiel medzi počtom existujúcich hrán medzi vrcholmi rovnakého typu a počtom takých hrán v náhodne vytvorenom grafe v pomeru ku všetkým existujúcim hranám. Vrcholy rovnakého typu sú tie, ktoré patria alebo majú patriť do rovnakej skupiny alebo triedy (komunity).

$$Q = \frac{1}{2m} \sum \sum_{i,j} \left[A_{ij} - \frac{d_i d_j}{2m} \right] \delta(c_i, c_j)$$

Def: Modularita [9]

5 Sociálna siet

Sociálna siet je množina sociálnych subjektov (uzly siete, spravidla jednotlivci alebo organizácie), ktoré sú prepojené jedným, alebo viacerými špecifickými druhmi vzájomnej závislosti, ako sú príbuzenstvo, priateľstvo, vzájomnosť, vízie, odpor, konflikt, obchod a pod. Sociálna siet z pohľadu teórie grafov je definovaná ako graf $G(V, E)$, kde V je množina entít (uzlov) a E je množina vzťahov (hrán) medzi týmito entitami.

Entity grafu môžu byť rôzne (základníci, jednotlivci, webové stránky, bankové účty, creditné karty, produkty). Nie je pravidlom, že len sociálna siet ako ju pozná mnoho ľudí je sociálnou sieťou aj formálne. Prvky sociálnej siete môžu ma napríklad aj skupina spolupracujúcich ľudí.

5.1 História sociálnych sietí

Pod pojmom sociálna siet si väčšina ľudí v dnešnej dobe predstaví služby ako *Facebook*, *Twitter* a pod. Tento pojem ale vznikol dlho pred vznikom internetu a dnešných sociálnych sietí. Prívlastok sociálny, ktorý sa v dnešnej dobe často vyniecha, je dôsledkom pôvodu analýzy sociálnych sietí. V druhej polovici 20. storočia sa simultánne v rôznych oblastiach skúmania vzťahov a chovania objavil nový pohľad na vzťahy medzi sociálnymi jednotkami a to ako na siet, graf. Preto prví predstavitelia analýzy sociálnych sietí boli pôvodne sociológovia alebo psychológovia (napríklad Moreno, Cartwright, Newcomb, Bavelas) a antropológovia (Barnes, Mitchell). Prvé použitie termínu "sociálna siet" sa pripisuje Barnesovi (1954).

V 30. rokoch 20. storočia psychiater Moreno rozvíjal sociometriu, predchodcu dnešnej analýzy sociálnych sietí. Vypytoval sa ľudí na priateľské vzťahy a skúmal, ako tieto vzťahy ovplyvňujú ich chovanie. Potom vynášiel (sám to tvrdil) tzv. *sociogram*, čo je diagram reprezentujúci ľudí ako body a vzťahy medzi ľuďmi ako úsečky, teda dnešnú sociálnu siet. Tento pojem sa ale začal používať až neskôr. Pomocou neho hľadal výrazné a izolované osoby v spoločnosti.

Zhruba o 20 rokoch neskôr antropológ Barnes začal skúmať, ako ovplyvnia vzťahy medzi ľuďmi nielen jednotlivcov, ale aj spoločnosť ako celok a zameral sa na štúdium skupín, komunít. Na práci Barnesa a jeho spolupracovníkov naviazala na Univerzite na Harvarde skupina vedená Harrisom Whitom. Tá začala budovať matematickú teóriu okolo dôležitejších pojmov zo sociálnych vied a umožnila tieto javy matematicky vyjadriť, merať a modelovať.

V druhej polovici 20. storočia sa rozšírilo povedomie o sociálnych sieťach a metódy sa začali používať aj v ďalších oboroch ako ekonómia, biológia, doprava atd.

5.2 Analýza sociálnych sietí

Analýza sociálnych sietí je interdisciplinárna veda s koreňmi v sociológii, psychológii, štatistike a teórie grafov. Analýza sociálnej siete chápe sociálnu siet ako systém prepojenia uzlov (individuálnych aktérov) prostredníctvom hrán (ich vzťahov). Možno teda povedať, že nadvázuje na matematickú teóriu grafov a metódy sietovej analýzy. Výsledkom analýzy môže teda byť mapa

graficky znázorňujúca všetky prvky skúmaného sociálneho systému a ich vzťahy (resp. vybrané charakteristiky jednotlivých vzťahov vyjadrené vhodným spôsobom graficky). Charakteristikou môže byť napríklad vzájomná sympatia či antipatia alebo pravidelná vzájomná komunikácia alebo spolupráca.

Analýza sociálnych sietí vystupuje napríklad ako základná technika v rámci modernej sociológie, antropológie, sociálnej lingvistiky, geografie, sociálnej psychológie, ekonómie a biológie rovnako ako populárna téma pre výskum.

5.3 Komunity v sociálnych sieťach

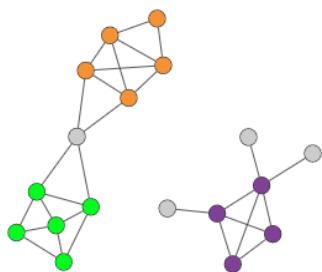
Sociálne siete sú riedke grafy zložené z hustých podgrafov. Tieto husté podgrafové sú nazývané komunity. Najčastejšia definícia komunity: *Komunita je zhľuk uzlov, kde počet vnútorných hrán v komunite je väčší ako počet vnokajších hrán – mimo komunity.* [10]

5.3.1 H2: Predpoklad súvislosti a hustoty

Komunity sú lokálne husto prepojené subgrafové v sieti. Toto očakávanie sa opiera o dva odlišné predpoklady:

Predpoklad súvislosti

Každá komunita odpovedá súvislému podgrafu, podobne ako subgrafové tvorené oranžovými, zelenými alebo fialovými uzlami na obrázku 7. V dôsledku toho, ak sa siet skladá z dvoch izolovaných komponent, každá komunita je obmedzená len na jednu komponentu. Táto hypotéza tiež naznačuje, že na tejto zložke sa komunita nemôže skladať z dvoch subgrafov, ktoré nemajú vzájomnú väzbu. V dôsledku toho oranžové a zelené uzly tvoria samostané komunity. [11]



Obr. 7: Komunity

Predpoklad hustoty

Uzly v komunite viac pravdepodobne združujú ďalších členov komunity než uzly v iných komunitách. Oranžové, zelené a fialové uzly toto očakávanie spĺňajú. [11]

Inými slovami, všetci členovia komunity musia byť dosiahnutelní cez ostatných členov tej istej komunity (súvislosť). V tom istom čase predpokladáme, že uzly, ktoré patria do komunity majú vyššiu pravdepodobnosť spájať ostatných členov tejto komunity ako uzly, ktoré do tejto komunity nepatria (hustota). [11]

5.3.2 Maximálne kliky

Jeden z prvých článkov o štruktúre spoločenstva publikovaný v roku 1949, definoval komunitu ako skupinu jednotlivcov, ktorej členovia sa navzájom poznajú [12]. V teoretických termínoch grafov to znamená, že komunita je komplexný subgraf alebo klika. Klika automaticky uspokojuje H2 - je to spojený subgraf s maximálnou hustotou väzieb. Aj keď zobrazenie komunít ako kliky má niekoľko nevýhod:

- Zatiaľ čo v sieťach sú časté trojuholníky, väčšie kliky sú vzácne.
- Požiadavka na to, aby komunita bola kompletnejší subgraf, môže byť príliš reštriktívna a chýba mnoho ďalších legitímnych komunít. [11]

5.4 Silné a slabé komunity

Zvažujme súvislý subgraf C s N_c uzlami v sieti. Vnútorný stupeň k_i^{int} uzla i je počet prepojení, ktoré sa pripojujú k iným uzlom v C . Externý stupeň k_i^{ext} je počet spojení, ktoré sa pripojujú k zbytku siete. Ak je $k_i^{ext} = 0$, každý sused i je vnútri C a preto C je dobrá komunita pre uzol i . Ak je $k_i^{int} = 0$, musí byť uzol priradený k inej komuniti. Tieto definície nám umožňujú rozlíšiť dva druhy spoločenstva.

Silná komunita

C je silná komunita, ak každý uzol vnútri C má viac spojení vo vnútri komunity ako s celou sieťou [13], [14]. Konkrétnie, podgraf C tvorí slabú komunitu ak pre každý uzol $i \in C$:

$$k_i^{int}(C) > k_i^{ext}(C)$$

Def: Silná komunita [11]

Slabá komunita

C je slabá komunita, ak celkový vnútorný stupeň subgrafa prekračuje svoj celkový externý stupeň [14]. Konkrétnie subgraf C tvorí slabú komunitu ak:

$$\sum_{i \in C} k_i^{int}(C) > \sum_{i \in C} k_i^{ext}(C)$$

Def: Slabá komunita [11]

5.5 Detekcia komunít

Detekcia komunít je proces identifikácie zhľukov uzlov siete silne prepojených medzi sebou a menej silne prepojených so zvyškom siete. Detekcia komunít v grafoch má za cieľ identifikovať moduly a ich prípadnú hierarchickú organizáciu.

Problém detektie komunít vyžaduje rozdelenie siete do komunít husto prepojených uzlov, pričom uzly patriace do odlišných komunít sú len slabo prepojené. Presné formulácie tohto optimalizačného problému sú známe ako výpočtovo neriešiteľné. Vyhľadávanie rýchlych algoritmov pritiaholo veľký záujem vďaka zvyšujúcej sa dostupnosti rozsiahlych sieťových dátových súborov a vplyvu sietí na každodenný život. Môžeme rozlišovať niekoľko typov algoritmov detektie komunít: *rozdeľovacie* algoritmy - tie detekujú spojenie vnútri siete a postupne ich odstraňujú zo siete, *algomeratívne* algoritmy - zlúčujú podobné uzly a postupne komunity podľa spoločných črt a *optimalizačné* metódy sú postavené na maximalizácii objektívnej funkcie. Kvalita rozdielov vplývajúcich z týchto metód sa často meria takzvanou modularitou. Je to hodnota v intervale od -1 do 1, ktorá meria hustotu spojov vnútri komunít v porovnaní s prepojeniami medzi komunitami.

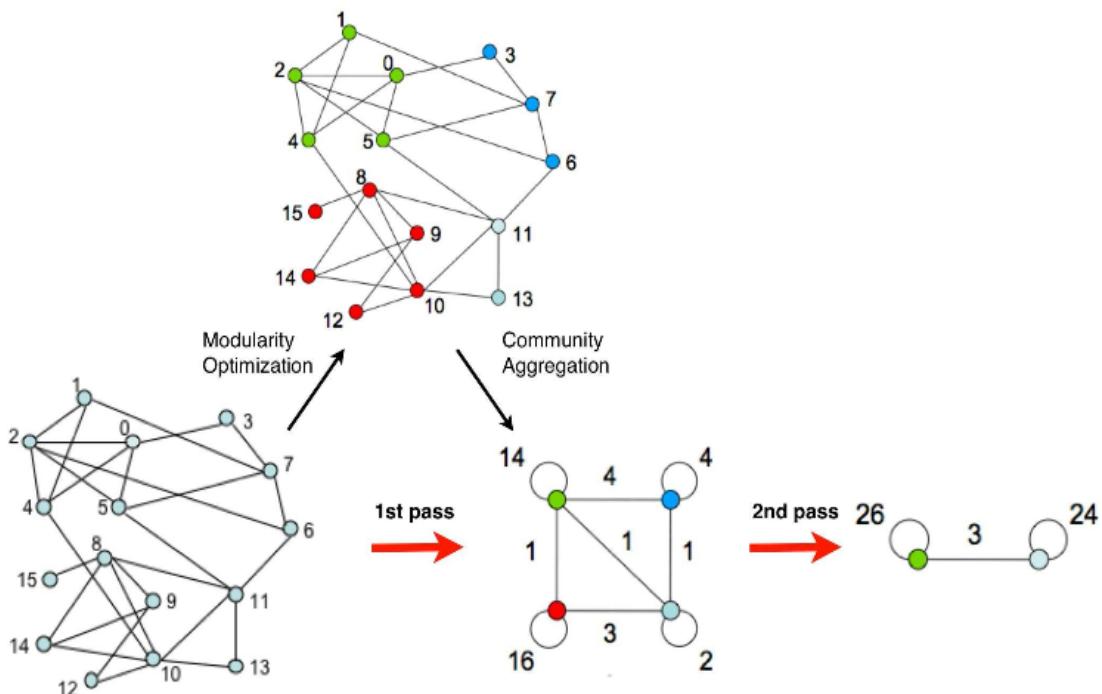
5.5.1 Louvainov algoritmus pre detekciu komunít

Veľmi oblúbeným a rýchlym algoritmom pre detekciu komunít je Louvainova metóda, ktorú navrhli Blondel, Guillaume, Lambiotte a Lefebvre [15]. Je to jednoduchá metóda pre extrakciu komunitnej štruktúry veľkých sietí. Je to heuristická metóda, ktorá je postavená na optimalizácii modularity. Je preukázané, že prekoná všetky ostatné známe metódy detektie komunít, pokiaľ ide o čas výpočtu. Navyše kvalita detekovaných komunít je veľmi dobrá.

Výpočet algoritmu je rozdelený do dvoch fáz, ktoré sa iteratívne opakujú. Predpokladajme, že začíname s váženou sieťou s N uzlami. Pokiaľ ide o neváženú siet, základná hodnota váhy je 1. Ako prvé označíme každý uzol siete inou komunitou. Takže v tomto prvotnom rozdelení je toľko komunít, ako je uzlov. Potom pre každý uzol i uvažujeme susedov j a vyhodnotíme prírastok modularity, ktorý by nastal, ak z sme odstránili uzol i z jeho komunity a priradili by sme ho do komunity uzla j . Uzol i je potom vložený do komunity, pre ktorú je tento prírastok najvyšší, ale len ak je tento prírastok kladný. Ak nie je možný žiadny kladný prírastok, uzol

i ostáva vo svojej komunite. Tento proces je aplikovaný opäťovne a sekvenčne pre všetky uzly kym sa nedosiahne žiadne zlepšenie a prvá fáza je kompletnej. Prvá fáza končí, keď je dosiahnuté lokálne maximum modularity, ked žiadny uzol už nemôže zlepšiť modularitu. Je taktiež dôležité, že výstup algoritmu záleží na postupe, v ktorom sú uzly brané do úvahy. Výsledky algoritmu ale naznačujú, že usporiadanie uzlov nemá významný vplyv na získanú modularitu. Zoradenie však môže ovplyvniť výpočtový čas. Problém pri výbere objednávky preto stojí za to študovať, pretože by mohol poskytnúť dobrú heuristiku na zvýšenie výpočtového času.

Druhá fáza algoritmu spočíva vo vytvorení novej siete, ktorej uzly sú komunity nájdené počas prvej fázy algoritmu. K tomu, aby sa nová sieť vytvorila, vähy spojení medzi novými uzlami sú dané sumou väh prepojení medzi uzlami korešpondujúcich dvoch komunit. Spojenia medzi uzlami tej istej komunity vedú k slučkám v novej sieti. Keď je druhá fáza kompletnej, je možné znova aplikovať prvú fázu algoritmu na výslednú váženú siet a proces opakovať. Pri konštrukcii sa počet komunit znižuje pri každom priechode. Proces sa opakuje, kým nie sú žiadne ďalšie zmeny a dosiahne sa maximálna modularita.



Obr. 8: Vizualizácia krokov Louvainovho algoritmu.

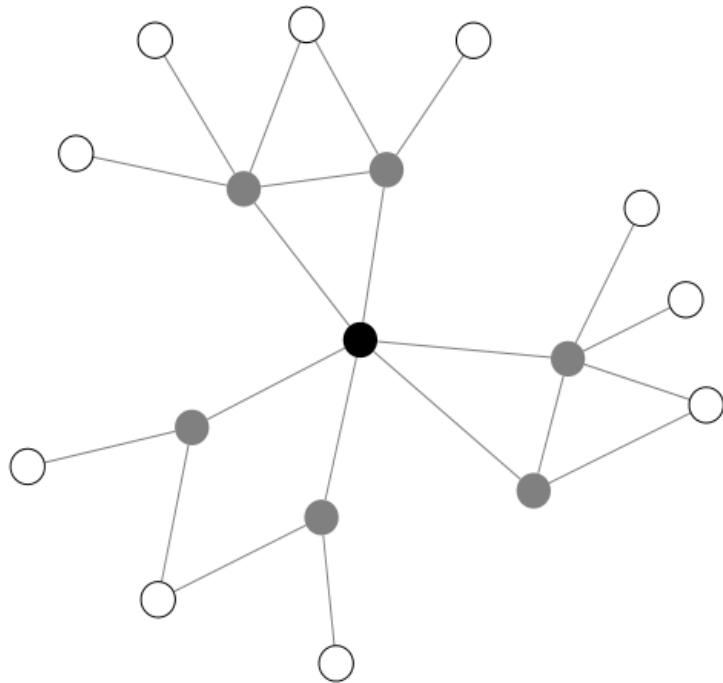
Každý priechod je tvorený dvomi fázami: prvá, kde je modularita optimalizovaná tým, že umožňuje len miestne zmeny komunit a druhá, kde nájdené komunity sú agregované tak, aby bolo možné vytvoriť sieť komunit. Priechody sú opakované iteratívne kým nie je možný žiadny nárast modularity.

5.6 Ego siet

Ego siet je siet tvorená uzlami, ktoré sa nazývajú aj *alter* uzlami, ktoré sa formujú okolo určitého uzla, ktorý sa nazýva *ego*. Toto ego sa niekedy zo siete vynecháva za účelom analýzy zmien siete. To záleží od danej analýzy [16]. Ego je individuálny ústredný uzol. Siet môže mať toľko ég, kolko má uzlov. Egá môžu byť osoby, skupiny, organizácie alebo spoločnosti.

5.6.1 Konštrukcia ego siete

V tejto práci konštruujem ego siet tak, že k uzlu, ktorý bol vybraný ako ego, sa pridajú hrany tak, aby spájal doposiaľ neprepojené komponenty siete. To môže pomôcť k analýze toho, či by sa daný človek hodil do vodcovskej pozície na základe jeho starých spojení. To, že sú k nemu pridané nové spojenia by bolo prirodzené, ak by sa daný jednotlivec dostal naozaj do vodcovskej (alebo inak egocentrickej) role. Príkladom môže byť napríklad to, keď sa v spoločnosti hľadá nový projektový manažér a z danej siete môžeme detektovať, či sa človek s danými vlohami nenachádza aj v aktuálnom tíme, ale na nižšej pozícii.



Obr. 9: Príklad ego siete.

6 Metódy analýzy sociálnych sietí

6.1 SSRM - Framework pre detekciu štrukturálnych rolí v sociálnych sieťach

Afra Abnar, Mansoureh Takaffoli, Reihaneh Rabbany, Osmar R. Zaiane [6] definovali *Structural social role mining framework*, ktorý je navrhnutý pre identifikáciu štrukturálnych rolí, pre identifikáciu zmien v sieti a analýzu dopadu zmien na sieť. Definujú základné sociálne roly v sieti(menovite Leader, Outermost, Mediator, Outsider).

6.1.1 Rola v kontexte SSRM

Sociálna rola je sice základný sociologický pojem, ale stále neexistuje žiadny konsenzus v jej definícii. Podľa SSRM je rola je považovaná za pozíciu jednotlivca v spoločnosti. Informácie o sociálnej sieti sú kategorizované do štrukturálnych a neštrukturálnych vlastností. Štrukturálne vlastnosti sú príbuzné ku konštrukcii grafu ako sú spojenia entít (hrany), štruktúra susedov a pozícia entity v tejto štruktúre. Ale neštrukturálne vlastnosti sú ostatné informácie, ktoré neodrážajú konštrukciu grafu ako atribúty entít a spojení. SSRM definuje rolu v sieti ako: Rola entity v sieti je to, ako sa entita správa voči ostatným a jej vplyv na atribúty a štruktúry ostatných entít.

6.1.2 Roly definované v SSRM

Ludské siete sú vnútorme zložené z viacerých komunít. V sociálnej sieti s viacerými komunitami, vlastnosti uzlov kolísu podľa toho, či je existencia komunít dostatočná alebo zanedbateľná. Z pohľadu sociálnej siete, uzol môže byť centrom celej siete, ale nie centrom v jeho komuniti. SSRM sa teda zameriava na štúdium sociálnych sietí s predpokladom existencie komunít v sieti, ako jej základnej črty.

V sociálnych sieťach môžu byť komunity explicitné alebo implicitné. Explicitné komunity sú postavené nezávisle na jej členoch a sú založené na množine pravidiel. V tomto prípade, ľudia sa stanú členmi tejto komunity častejšie až po zformovaní komunity. Zamestnanci firmy alebo študenti sú príkladom dvoch explicitných komunít. Zatiaľ čo formácia implicitných komunít tažko závisí na jej členoch a spojeniach. Tým pádom neexistuje žiadna externá podmienka na vybudovanie implicitnej komunity. Implicitné komunity sú postavené postupne ako sa ľudia spoločne stretávajú. Napríklad, skupina priateľov, v ktorej nie je žiadne pravidlo pre správanie sa jednotlivcov, je príklad implicitnej komunity. V oboch prípadoch explicitnej aj implicitnej komunity, by mali existovať aj špeciálne jednotlivci, ktorí tieto komunity manažujú a kontrolujú. Napríklad v školskej triede je to učiteľ alebo inštruktor. Pre firmu to je manažér vo vedení a pre skupinu priateľov je to zase človek, ktorého komunikačné schopnosti prinášajú ďalších členov alebo posilňujú vzťahy medzi tými stálymi. Títo dôležití jednotlivci sú ešte viac výrazní, keď je komunita obrovská.

Podľa toho SSRM framework definuje pre jednotlivcov v sociálnej sieti určité roly podľa ich vzťahov a pozícii v komunitách až po ich interakcie s ostatnými jednotlivcami. Z perspektívy komunít, v sieti existujú jednotlivci niekoľkých typov:

- so žiadnym vzťahom ku nejakej komunité
- so spojením s viacerými komunitami
- dôležitý členovia komunity
- bežný členovia komunity, ktorí formujú väčšinu
- nedôležitý členovia komunity, ktorí nemajú na komunitu pozorovateľný efekt

Na základe týchto poznatkov SSRM definuje štyri základné roly - **leader**, **mediator**, **outermost** a **outsider**.

6.1.2.1 Leader

Sú mimoriadni jednotlivci v zmysle centrality alebo významu v každej komunité. V reálnom svete bývajú títo členovia siete veliteľmi, riaditeľmi, manažérmi, vládcami, prezidentami, autoritami, administrátormi atď.

6.1.2.2 Outermost

Je to časť menej dôležitých jednotlivcov v každej komunité, ktorých vplyv a efekt na komunitu sú nižšie ako vplyv väčšiny členov komunity. Miesta, kde sa môže outermost v sieti nachádzať sú periféria alebo hranice grafu.

6.1.2.3 Mediator

Sú to jednotlivci, ktorí zohrávajú dôležitú rolu v spojení komunít v medzi sebou. Fungujú ako mosty medzi odlišnými komunitami. Do tejto skupiny patria vyjednávači, sprostredkovatelia alebo aj rozbočovače v sieti.

6.1.2.4 Outsider

Sú to jednotlivci, ktorí nie sú spojení so žiadnou komunitou v sieti. Bud majú takmer rovnaké prepojenie k rôznym komunitám alebo majú len veľmi slabé väzby na komunity.

6.2 Identifikácia štrukturálnych sociálnych rolí

Majúc sieť s komunitami explicitne známymi alebo extrahovanými nejakým dolovacím algoritmom, následne popisujem metodológie pre identifikovanie definovaných štrukturálnych rolí.

6.2.0.1 Outsider

Najviac priamočiarou rolou pre identifikáciu je outsider. Je to jednotlivec, ktorý v sieti nepatrí do žiadnej komunity. Identifikácia tejto roly je tak celkom priamočiara.

6.2.1 Leader

Leader je v každej komunite výnimočný centrálny člen. Pre identifikovanie takýchto uzlov SSRM využíva metriku *closeness centrality*.

6.2.1.1 Closeness centrality (Centralita blízkosti)

V súvislom grafe closeness centrality uzlu je metrika centrality v sieti, vypočítaná ako súčet dĺžok najkratších cest medzi uzlom a všetkými ostatnými uzlami v grafe. Čiže čím viac je uzol centrálnejší, tým bližšie je k ostatným uzlom.

$$C_i = \frac{1}{l_i} = \frac{n}{\sum_j d_{ij}}$$

Def: Closeness centrality

Pre každý uzol sa stanoví hodnota closeness centrality. Hodnoty closeness centrality sú blízke notmálnemu rozdeleniu, v ktorom 95% populácie dát patrí do intervalu $[\mu - 2 \cdot \sigma, \mu + 2 \cdot \sigma]$

Leadri ležia na hornom chvoste distribučnej funkcie, a teda horný interval použijeme pre identifikovanie leadrov. A teda uzly, ktoré majú väčšiu hodnotu closeness centrality ako krajná hodnota tohto intervalu, sú identifikovaní ako leadri.

6.2.2 Outermost

Podobne ako pri role *Leader* pre identifikovanie outermostov sa využíva metrika closeness centrality. Outermosti budú ležať však na spodnom chvoste distribučnej funkcie closeness centrality.

A tak teda uzly, ktoré majú hodnotu closeness centrality nižšiu ako $[\mu - 2 \cdot \sigma]$, sú outermosti.

6.2.3 Mediator

Rolu mediator zastávajú tí jednotlivci, ktorí spájajú viacero komunit a sú tzv. spojmy medzi komunitami.

Pre identifikáciu mediátorov sa definujú metriky založené na metrike betweeness centrality a to: *LBetweeness - LBC* a *CBetweeness - CBC* a ďalej metriky, ktoré vyjadrujú koľko rozdielnych komunit uzol spája: *DSCount* a *DSPair*.

6.2.3.1 LBeweeness

LPath - Pred definíciou LBeweeness je potrebné definovať LPath a to nasledovne: *LPath* je množina všetkých najkratších cest medzi lídrami dvoch rozdielnych komunit.

$$LPath = l | startNode(l) \in leaderSet(c_i) \wedge endNode(l) \in leaderSet(c_j) \wedge c_i \neq c_j$$

Def: Lpath

LBetweenss centralita pre uzol v - $LBX(v)$ je počet jedinečných LPath ktoré obsahujú v . Ak pre každú cestu $p \in LPath$ definujeme $I_l(p, v) = 1$ ak v leží na p , inak $I_l(p, v) = 0$ potom:

$$LB(v) = \sum_{p \in LPath} I_l(p, v)$$

Def: LBetweenss

6.2.3.2 CBetweenss

CBetweenss počíta počet najkratších ciest medzi rozdielnymi komunitami. s_p a e_p označujú štartovací a koncový uzol najkratšej cesty p . Taktiež c_v označuje komunitu, do ktorej uzol v patrí. Množina všetkých najkratších ciest, ktoré spájajú rozdielne komunity: $CPaths = \{p | c_{s_p} \neq c_{e_p}\}$. Taktiež definujeme $I_p(p, v) = 1$ ak v leží na ceste p a 0 keď neleží.

$$CB(v) = \frac{1}{2} \sum_{p \in CPaths} I_p(p, v)$$

Def: CBetweenss

6.2.3.3 Normalizovaná verzia CBetweenss

Pravdepodobnosť nájdenia viac viditeľných mediátorov vo väčších komunitách je väčšia v porovnaní s menšími komunitami. Táto situácia sa stáva, pretože vo väčších komunitách je pochopiteľne viac uzlov, čo vedie k viacerým najkratším cestám medzi nimi. Pre kompenzáciu tohto efektu je definovaná normalizovaná verzia CBC :

$$NBC(v) = \frac{1}{2} \sum_{p \in CPaths} \frac{I_p(p, v)}{\min(|c_{s_p}|, |c_{e_p}|)}$$

Def: Normalizovaná verzia CBetweenss

Navrhnuté metriky CBC a LBC sú nevyhnutné pre identifikovanie mediátorov, ale nie sú dostatočné. Napríklad pre sieť pozostávajúcu z desiatich komunít a dvoch mediátorov M_1 a M_2 , kde oba ležia na sto najkratších cestách medzi komunitami majú oba rovnaké hodnoty CBC . Kdežto M_1 spája dve rozdielne komunity, kým M_2 spája všetkých 10. Pri takomto scenárii M_2 spája komunity viac globálne a mal by byť skôr posudzovaný ako mediátor ako M_1 . A tak

SSRM definuje tzv. metriku **skóre rozmanitosti**, ktorá označuje rozdielne komunity, ktoré sú prepojené cez uzol.

6.2.3.4 Skóre rozmanitosti

Táto metrika ukazuje koľko rozdielnych komunit je spojených cez špecifický uzol v . Túto metriku definujeme v dvoch variantach:

1. **DSCount** - je definovaný ako počet rozdielnych komunit, ktoré sú spojené daným uzlom.

Nech $I_d(c_i, v) = 1$ ak $\exists p \in CPaths : s_p \in c_i \wedge v \in p$. Potom DCount uzla v je definovaný ako:

$$DS_{count}(v) = \frac{1}{2} \sum_{c_i} I_d(c_i, v)$$

Def: DCount

2. **DSPair** - Skóre rozmanitosti môže byť definované ako počet párov komunit, ktoré majú najmenej jednu najkratšiu cestu medzi ich členmi, ktoré prechádzajú uzlom v . Definujeme $I_d(c_i, c_j, v) = 1$ ak $\exists p \in CPaths : s_p \in c_i \wedge e_p \in c_j \wedge v \in p$

$$DS_{pair}(v) = \frac{1}{2} \sum_{c_i} \sum_{c_j \neq c_i} I_d(c_i, c_j, v)$$

Def: DSPair

Aj keď viac mediátorov môže mať rovnaké hodnoty jednotlivých metrík, môžu sa odlišovať napríklad v počte komunit, ktoré spájajú. *SSRM* to berie do úvahy a definuje tzv. *mediacy score* ako násobok normalizovanej CBetweenss a skóra rozmanitosti:

$$MS(v) = NCB(v) \cdot DS_{count}(v)$$

Def: Mediacy score

6.3 Brokerage roly

Jednoducho povedané, *brokerage* sa vyskytuje tam, keď jeden aktér siete poskytuje most medzi dvoma inými aktérmi, ktorí medzi sebou inak prepojení nie sú. Koncept *brokerage* rôl bol použitý vo veľmi veľa iných kontextoch, záleží len na jeho formalizácii. Aj keď je *brokerage* tradične konceptualizovaný ako dynamický fenomén, identifikácia *brokerage* rôl sa často využíva aj v oblasti statických spoločenských vzťahov.

Jedným známym kontextom pre *brokerage* je prípad obchodných vzťahov. V tomto prostredí, tito jednotlivci alebo organizácie, politické entity, ktorí boli schopní previezť tovar z jedného

miesta na druhé a kontrolovať ich rozšírenie, zohrávali kľúčovú rolu v udržiavaní obchodu na regionálnej a kontinentálnej úrovni. S prostredkováním kontaktov medzi vzdialé tretie strany (ktoré si nemôžu vymeniť informácie inak), títo aktéri povolili uvoľnenie kritických, priestorovo lokalizovaných zdrojov naprieč roziahlym územím, čo usnadňovalo rast zložitejších spoločností. Kým *brokerage* vo výmenných sieťach má dôležité systematické následky, jeho efekt na individuálnej úrovni bol oceňovaný viac intenzívne sociálmi (napr. v [17] [18] [19]).

Je zrejmé, že *brokerage* sa môže vyskytnúť v mnohých nastaveniach a povaha *brokerage* procesu samotného sa líši od kontextu. Vširšom zmysle tento proces spadá pod tri triedy - *transfer brokerage*, v ktorom *broker (ego)* vede informácie a iné zdroje od jedného jednotlivca k druhému, ktorí nie sú priamo prepojení. Potom *matchmaking brokerage*, v ktorom ego predstavuje alebo inak umožňuje spojenie jedného jednotlivca k druhému a nakoniec *coordination brokerage*, v ktorom ego usmerňuje kroky ostatných a tak vyriešia svoje závislosti bez toho, aby museli byť priamo prepojení.

Brokerage je stav alebo situácia, v ktorej účastník spája inak neprepojených účastníkov alebo zapĺňa medzery alebo diery v sieti. [17] Na obrázku je *broker* alebo aj *sprostredkovateľ* zastúpený čiernym uzlom, ktorý vyplňuje dieru v sieti alebo spojuje ostatných jednotlivcov reprezentovaných bielymi uzlami, ktoré predtým neboli navzájom prepojené priamo.



Obr. 10: Príklad brokerage procesu

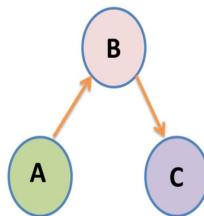
Broker môže prepojiť oddelené oblasti siete sociálnymi, ekonomickými alebo politickými aspektami a preto je jediný, kto má prístup k cenénym informáciám a zdrojom z rôznych oblastí siete. *Brokerage* je mechanizmus, ktorý umožňuje izolovaným či neprepojeným členom siete zdieľať informácie a zdroje a ekonomicky, politicky či spoločensky ovplyvňovať. [20]

Práve kvôli spojeniu a kontrole nad jedinečnými informáciami a zdrojmi medzi neprepojenými účastníkmi siete má aktér, ktorý zohráva rolu sprostredkovateľa (*broker*) v sieti väčší prístup k informáciám a zdrojom v porovnaní s tými, ktorí sprostredkovateľmi nie sú. *Broker* (sprostredkovateľ) môže fažiť z tejto kontroly nad informáciami a zdrojmi, môže sa stat silnejší v sieti a môže vykazovať zvýšenú efektivitu vo svojej práci. [20]

Detailnejšiu kategorizáciu *Brokerage* rôl predstavili Gould a Fernandez [17], kde predstavili koncept *brokerage* typológie. Táto typológia delí *brokerage* do piatich typov na základe smeru toku informácií - tokov v sieti - a rozdeľuje aktérov do vzájomne sa vylučujúcich skupín, tried alebo organizácií. Typy sprostredkovateľov sú *liaison*, *itinerant*, *coordinator*, *gatekeeper* a *representative*.

6.3.1 Liaison

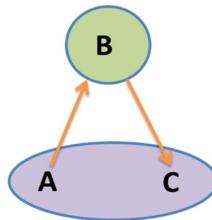
Liaison brokerage je *broker* spojenie medzi dvoma rozdielnymi skupinami, do ktorých on nepatrí. Na obrázku je *broker* (B) spojený s dvoma skupinami (A a C), ale nie je súčasťou ani jednej tejto skupiny a teda tvorí spojenie medzi aktérom A a aktérom C.



Obr. 11: Liaison brokerage

6.3.2 Itinerant

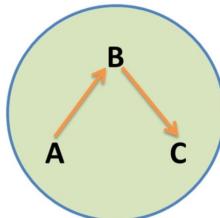
Pri tomto type *brokerage* roly dvoja neprepojení aktéri (A a C) patria do jednej skupiny, kým *broker* (B) patrí do inej skupiny. *Itinerant broker* je tiež nazývaný *consultant broker*, pretože *broker* sa chová ako konzultant pre oboch nespojených aktérov tej istej skupiny.



Obr. 12: Itinerant brokerage

6.3.3 Coordinator

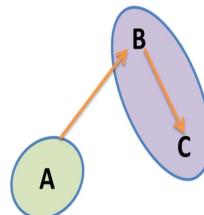
V role *coordinator* všetci traja aktéri patria do rovnakej skupiny a sprostredkovanie informácií a zdrojov sa deje v rámci skupiny.



Obr. 13: Coordinator brokerage

6.3.4 Gatekeeper

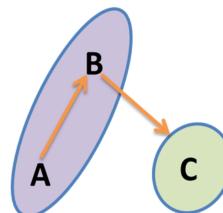
V tomto typie *brokerage* roly *broker*(B) a jeden z dvoch neprepojených aktérov (C) patria do jednej skupiny, kým iný neprepojený aktér (A) patrí do rozdielnej skupiny. *Broker* tohto typu kontroluje prichádzajúce informácie a zdroje v rámci jeho skupiny a robí rozhodnutia a tom, či majú alebo nemajú neprepojení aktéri v skupine prístup k informáciám a zdrojom.



Obr. 14: Gatekeeper brokerage

6.3.5 Representative

Representative rola je podobná role *gatekeeper* role, *broker* (B) a jeden nespojený aktér (A) patra do jednej skupiny kým ten druhý nespojený aktér (C) patrí do inej rozdielnej skupiny, ale smer toku informácií alebo zdrojov je rozdielny.



Obr. 15: Representative brokerage

6.3.6 Identifikácia brokerage rolí

Päť typov *brokerage* rôl reprezentujú unikátne sociálne roly zapuzdrujúce elementárny aspekt aktérovej štrukturálnej pozície v danej sieti. Jeden jednotlivec však môže zohrávať viac *brokerage* rolí naraz. Preto Gould & Fernandez [15] kvantifikovali celkovú participáciu jednotlivca v *brokerage* rolách pomocou *brokerage* skóra. Formálne definovali *brokerage* v grafe reprezentujúcim asymetrickú reláciu R : Nech a je *broker* medzi b a c iba ak bRa , aRc a $a\bar{R}c$, kde bRa indikuje, že b je prepojené s a reláciou R a $b\bar{R}c$ je negácia bRc . S touto definíciou, *brokerage* skóre sa vypočíta súčtom počtu kôľko krát táto podmienka platí pre špecifickú kombináciu spojenia aktérov. To znamená, že ak nejaký aktér x zohráva pozíciu *coordinator* dva krát a pozíciu *representative* tri krát, tak aktér bude mať skóre pre pozíciu *coordinator* = 2, pre pozíciu *representative* = 3 a jeho celkové *brokerage* skóre bude 5.

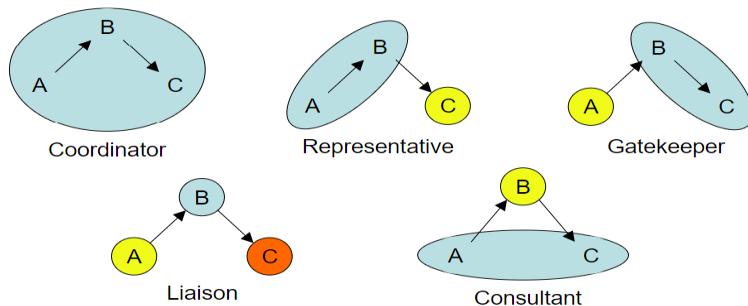
Formalizácia *brokerage* rolí podľa Goula a Fernandeza je definovaná pre siete, v ktorých sú spojenia (hrany) orientované, čiže reprezentujúce vzťahy, pre ktoré môžeme rozlíšiť odosielateľa a prijímateľa. Kedže v mojej koncepcii siete, kde jednotlivé uzly sú členovia tímu a hrana medzi nimi je práve vtedy, keď medzi nimi prebehla konverzácia, moja vytvorená sieť je neorientovaná. Zovšeobecnenie na neorientovanú sieť je celkom jasné; s takýmito dátami, každá hrana je považovaná za obojsmernú. Aj keď toto prináša jednu dôležitú zmenu originálnej formalizácie: v prípade neorientovaných vzťahov nemôžeme rozlíšiť rolu *gatekeeper* od roly *representative*, pretože neprítomnosť obojsmerných vzťahov redukuje tieto dve roly do jednej a *brokerage* skóre bude pre tieto dve roly identické. [21]

6.3.7 Popis metódy pre identifikáciu brokerage rolí

Podmienka pre detekovanie brokerage rolí je prítomnosť komunit. Pre každý uzol grafu si získam jeho susedné uzly. Tento uzol označujem ako *B* uzol. Potom prechádzam jeho susedné uzly každý s každým a kontrolujem, do akej komunity daný uzol patrí. Počas týchto prichodov označujem tieto uzly ako uzol *A* a uzol *C* a vyhodnocujem nasledovné podmienky:

1. Pokiaľ medzi uzlami *A* a *C* existuje hrana, preskočím ich a začínam prichod znova. Ak medzi nimi hrana nie je, prechádzam na podmienku 2.
2. Pokiaľ uzly *B* a *A* a *C* nepatria do jednej spoločnej komunity, uzol *B* je identifikovaný ako *Liaison* a navýši sa jeho skóre pre túto rolu.
3. Pokiaľ sú uzly *A* a *C* v rovnakej komunite a uzol *B* je v rozdielnej komunite, uzol *B* je detekovaný ako *Itinerant (Consultant)* a navýši sa jeho skóre pre túto rolu.
4. Pokiaľ sú uzly *A* a *B*, *C* v rovnakej komunite a uzol *C* je v inej komunite alebo ak sú uzly *B* a *C* v rovnakej komunite a uzol *A* je v inej komunite, uzol je detekovaný ako *Representative* a zároveň ako *Gatekeeper*, pretože nerozlišujem smer spojenia uzlov a teda im skóre navýšim zhodne.

Jeden uzol môže byť identifikovaný aj všetkými *brokerage* rolami. Ked je prichod všetkými uzlami dokončený, spočíta sa celkové skóre súčtom čiastkových skóre pre každú rolu.



Obr. 16: Identifikácie *brokerage* rolí [1]

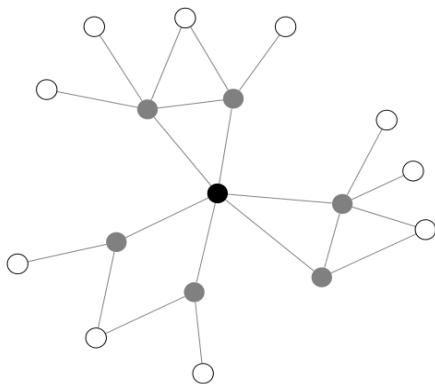
6.4 Analýza ega

Analýza ego sietí sa stáva stále viac dôležitou s rastom sietí. Je oveľa jednoduchšie v obrovských sietach analyzovať ego a jeho okolie ako celú sieť ako celok. Napríklad ak jeden človek má priemerne 5 blízkych osôb, potom v meste s populáciou desať tisíc ľudí bude päťdesiat tisíc priateľských väzieb. A ak by sme chceli študovať známosti? Riešením by bol výber podmnožiny obyvateľov mesta a ich *alter* uzlov.

Ďalšou odpovedou na otázku, prečo študovať ego siete, je to, že niekedy nás nezaujíma sieť ako celok alebo komunity a podobne, ale zaujímajú nás dôležití alebo inak zaujímaví jednotlivci (lídri, umelci, tínedžeri a pod.) Siet ega je zaujímavá, pretože je zdrojom informácií, sociálnej podpory, prístupu ku zdrojom, vplyvu a ďalších faktorov.

6.4.1 Veľkosť ego siete

Veľkosť ego siete je jednoduchá, ale veľavravná charakteristika. Definuje ju stupeň ego uzla, alebo teda počet *alter* uzlov ega. Hovorí o sociálnej podpore, prístupu k informáciám a zdrojom.



Obr. 17: Veľkosť ega - stupeň uzla: 6

6.4.2 Kompozícia ego siete

Čo sa týka kompozície ego siete, môžeme sledovať podobnosť medzi egom a jeho *alter* uzlami. Pre reprezentáciu podobnosti sa používa *homofília*. Môžeme predpokladat, že existuje vzťah medzi nejakým javom a tým, či ego zdiela so svojimi *alter* uzlami nejakú vlastnosť (profesia, vzdelenie a pod.) Napríklad je prirodzené, keď niekto, kto zastáva pozíciu CFO (Chief Financial Officer) je obklopený ľuďmi, ktorí riešia financie alebo napríklad politici bývajú obklopení členmi rovnakej politickej strany.

Pre identifikáciu homofilie som využila prítomnosť komunit v sieti a použila som *Krackhardt-Sternov E-I index* [22].

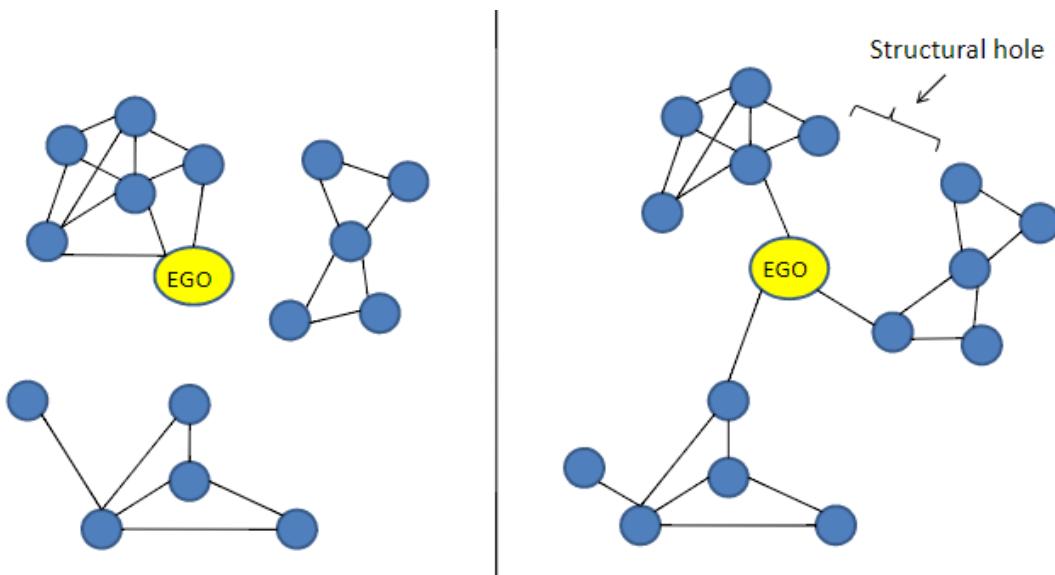
$$\frac{E-I}{E+I}$$

- E je počet spojení s členmi inej skupiny (komunity) I je počet spojení s členmi rovnakej skupiny (komunity)

- nadobúda hodnoty od -1(homofilia) do +1(heterofilia)

6.4.3 Štruktúra ego siete

Štrukturálna analýza sa opiera o informácie, či existujú alebo neexistujú spojenia medzi *alter* uzlami ego uzla. Princíp spočíva v tom, že nedostatok spojení medzi *alter* uzlami môže priniesť určité benefity samotnému egu. Tento princíp sa v analýze sociálnych sietí nazýva princíp štrukturálnych dier (ang. *structural holes*). Medzi benefity, ktoré prinášajú štrukturálne diery egu patria prístup k novým informáciám, moci alebo k slobode.

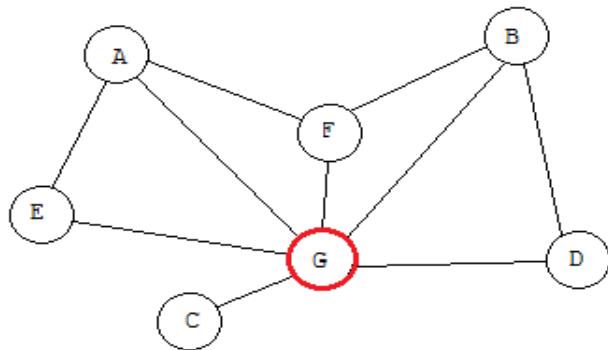


Obr. 18: Málo štrukturálnych dier vs. veľa štrukturálnych dier.

Koncept štrukturálnych dier je koncept analýzy sociálnych sietí vyvinutý R. S. Burтом. Predstavil tento pojem v snahe vysvetliť vznik rozdielov v sociálnom kapitále. Burtova teória naznačuje, že jednotlivci majú isté výhody alebo nevýhody podľa toho, ako sú zakotvené v spoločenských štruktúrach. Štrukturálna diera je chápana ako medzera medzi dvoma jednotlivcami (chýbajúca hrana medzi uzlami), ktorí majú doplňujúce zdroje informácií. [23]

6.4.3.1 Efektívna veľkosť

Burt predstavil mieru redundancie siete, Borgatti vyvinul zjednodušenú verziu efektívnej veľkosti pre nevážené siete [24]. Redundancia = $\frac{2t}{n}$, kde t je počet všetkých spojení v egocentrickej sieti (s výnimkou spojení k egu) a n je počet všetkých uzlov v egocentrickej sieti (s výnimkou ega). Táto formula môže byť modifikovaná pre výpočet efektívnej sily ego siete. Efektívna veľkosť ego siete = $n - \frac{2t}{n}$ (rozdiel počtu *alter* uzlov ega a sumy ich redundancií = $6 - 1.33 = 4.67$). Efektívna veľkosť udáva počet neredundantných uzlov ego siete.



Uzol G je ego	A	B	C	D	E	F	Celkom
Redundancia	3/6	2/6	0/6	1/6	1/6	1/6	1.33

Obr. 19: Príklad výpočtu redundancie

Čím viac je každý uzol odpojený od ostatných primárnych kontaktov, tým vyššia bude efektívna veľkosť. Tento indikátor nadobúda hodnoty od 1 (sieť poskytuje len jediné spojenie (hranu)) až do celkového počtu spojení, kedy každý kontakt (alter) je neredundantný.

7 Aplikácia

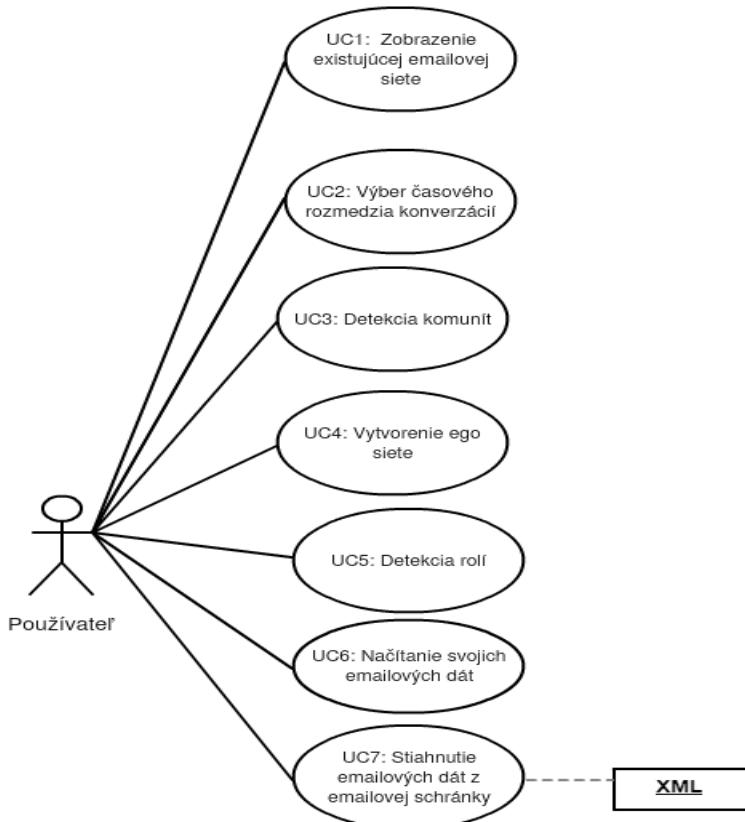
Táto kapitola obsahuje všetky podrobnosti o vývoji aplikácie, návrhu a ďalej špecifikáciách požiadavkov. Sú tu uvedené informácie o implementácii, návrhu, návrhových vzoroch, ale aj konštrukcii siete, predpríprave dát. Táto časť taktiež obsahuje diagramy najdôležitejších tried aplikácie alebo diagramy prípadov použitia.

7.1 Špecifikácia

Aplikácia slúži ako užívateľské rozhranie na analýzu emailovej komunikácie a vizualizáciu analytických výstupov. Aplikácia umožňuje exportovať dátá z emailovej schránky alebo importovať vlastný XML súbor s emailovými dátami a ďalej s týmito dátami pracovať a zobrazovať siet emailovej komunikácie. Umožňuje vytvorenie ego-siete alebo detektovať vo vytvorennej siete komunity. Najdôležitejšou časťou aplikácie je detekcia štrukturálnych rolí v sieti, čiže detekcia dôležitých a nedôležitých členov emailovej komunikácie.

7.1.1 Funkčné požiadavky

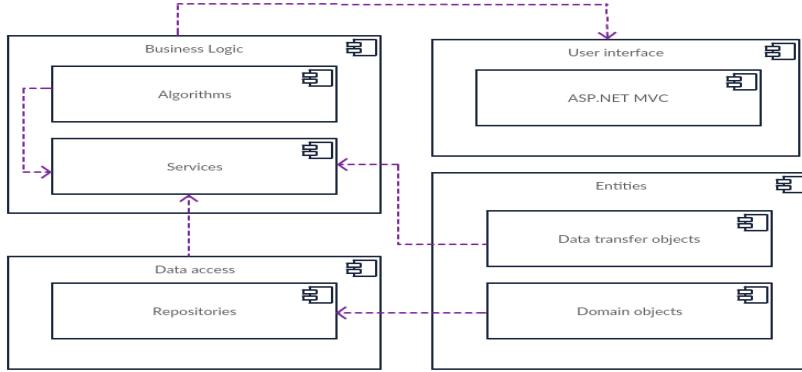
- Export dát z emailovej schránky
- Import vlastného XML súboru s emailovými dátami
- Zobrazenie informácií o emailovej sieti
- Vizualizácia emailovej siete
- Vytvorenie ego-siete
- Detekcia komunít
- Detekcia štrukturálnych rolí v sieti
- Výber časového rozmedzia emailových konverzácií



Obr. 20: UseCase Diagram

7.2 Návrh

Aplikácia je vytvorená ako .NET aplikácia (veria .NET Frameworku 4.6). Je vytvorená ako trojvrstvová, pre uloženie dát sa používa SQL databáza. Najnižšia vrstva aplikácie slúži na získavanie dát z databázy, pre prepojenie s databázou a posielanie dát z aplikácie do databázy používam Entity Framework a používam tu návrhový vzor Repository. Od tejto časti je oddelená časť s business logikou a na najvyššej časti, ktorá slúži len na zobrazenie dát a komunikáciu s užívateľom, používam známy prístup Model-View-Controller.

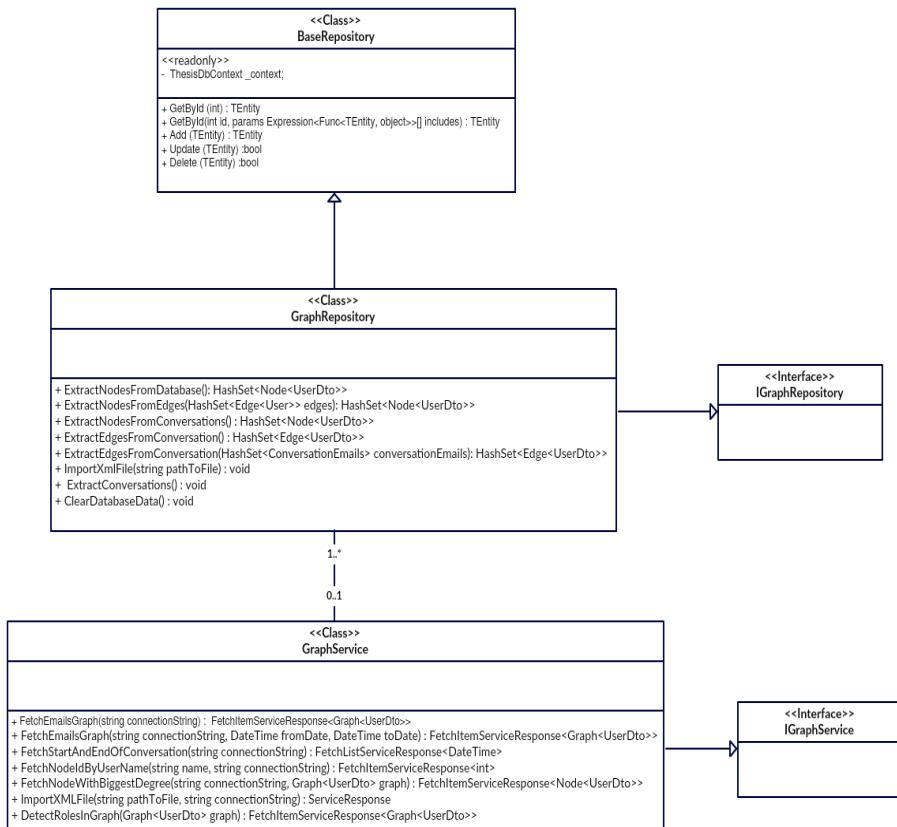


Obr. 21: Diagram komponent znázorňujúci jednotlivé komponenty architektúry aplikácie

7.2.1 Návrhové vzory

Repository

Návrhový vzor Repository je základným kameňom doménou riadeného návrhu. Model aplikácie teda nemá poňatie o tom, akým spôsobom je perzistovaný. O to sa stará práve Repository. Naviac práve vďaka tomu, že sa o persistenciu stará cudzí objekt, stačí poznať len jeho rozhranie a v prípad potreby ho ľahko nahradíť iným. [25]

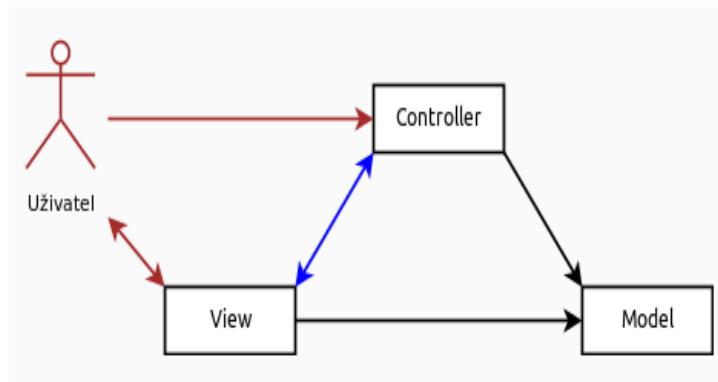


Obr. 22: Triedny diagram - Repository pattern

Model-View-Controller V aplikácii je použitý tradičný vzor Model View Controller (MVC). Je to jeden z najpoužívanejších a najobecnejších architektonických vzorov.

MVC rozdeľuje program do troch hlavných častí:

- **Model** - dátá a súvisiace operácie
- **View** - prezentácia dát (užívateľské rozhranie), obsahuje priamy odkaz na model, aby mohol jeho dátá prezentovať vonkajšiemu svetu
- **Controller** - riadi tok udalostí v programe, konkrétnie v tejto aplikácii kontrolery obsahujú len volanie metód z inej vrstvy aplikácie



Obr. 23: Model-View-Controller

7.3 Dôležité rozhodnutia

Pri navrhovaní aplikácie bolo potrebné urobiť niekoľko dôležitých rozhodnutí.

7.3.1 Dostupnosť dát

Pôvodne sa zvažovalo použitie aplikácie a analýzy dát nad verejne dostupnou anonymizovanou emailovou sadou. Emailových dát je ale veľmi málo a chcela som, aby sa výsledky práce dali overiť nie len inými analytickými nástrojmi, ale aj empiricky. Takže som využila to, že pracujem a moja emailová schránka teda nie je chudobná na maily. Navyše mi radi pomohli aj moji kolegovia a poskytli mi svoje emailové dátá. Takto som zozbierala reálne emailové dátá štyroch ľudí, o ktorých je známe ich postavenie v týme alebo aj dátum nástupu do práce. Takže výsledky daných algoritmov som vedela porovnať s reálnou situáciou v kolektíve.

7.3.2 Webová vs. desktopová aplikácia

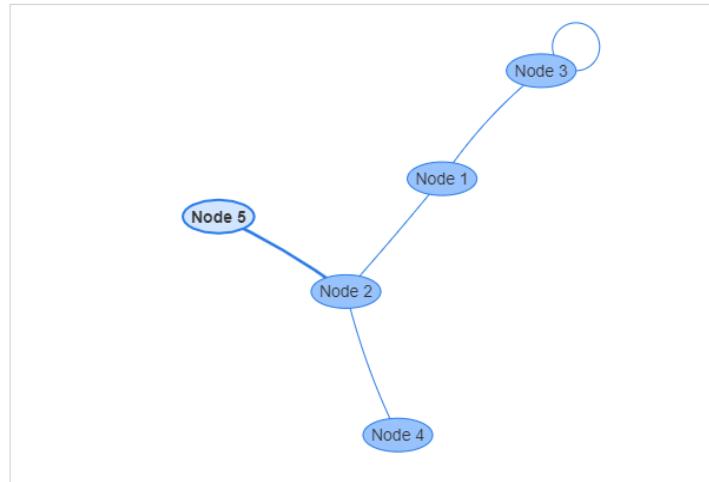
Bolo nutné sa rozhodnúť, či vyvíjať aplikáciu ako webovú alebo desktopovú. Ako platforma bola zvolená Microsoft .Net a programovací jazyk C#. Jednou z variant bola desktopová aplikácia

vyvíjaná vo Windows Forms. WinForms je osvedčná technológia, je vyladená, v základe obsahuje veľké množstvo grafických prvkov. Toto sú výhody rozšírenej a dlho používanej technológie. Nevýhoda je ale práve zastaralosť a fažkopádnosť v kreslení a spravovaní grafického rozhrania. Keďže ale doba ide dopredu a web a webové aplikácie sú stále viac používanejšie a v súčasnosti existuje mnoho grafických knižníc pre vizualizáciu grafického rozhrania, rozhodla som sa aplikáciu vyvíjať ako webovú.

7.4 Použité knižnice

Vis.js

Vis.js je dynamická vizualizačná knižnica. Knižnica je navrhnutá tak, aby bola ľahko ovládateľná a aby mohla spracovať veľké množstvo dynamických dát a umožňovala manipuláciu s dátami a interakciu s nimi. Knižnica sa skladá z časti *DataSet*, *Timeline*, *Network*, *Graph2d* a *Graph3d*. Pre moju aplikáciu som používala len časť *Network*.



Obr. 24: Jednoduchá sieť vytvorená s použitím knižnice vis.js

```

<style type="text/css">
  #mynetwork {
    width: 600px;
    height: 400px;
    border: 1px solid lightgray;
  }
</style>

<script type="text/javascript">
  // create an array with nodes
  var nodes = new vis.DataSet([
    {id: 1, label: 'Node 1'},
    {id: 2, label: 'Node 2'},
    {id: 3, label: 'Node 3'},
    {id: 4, label: 'Node 4'},
    {id: 5, label: 'Node 5'}
  ]);

  // create an array with edges
  var edges = new vis.DataSet([
    {from: 1, to: 3},
    {from: 1, to: 2},
    {from: 2, to: 4},
    {from: 2, to: 5},
    {from: 3, to: 3}
  ]);

  // create a network
  var container = document.getElementById('mynetwork');
  var data = {
    nodes: nodes,
    edges: edges
  };
  var options = {};
  var network = new vis.Network(container, data, options);
</script>

```

Obr. 25: Príklad použitia knižnice vis.js

LouvainSharp

Ďalšou použitou knižnicou bola knižnica *LouvainSharp*, ktorá implementuje Louvainov algoritmus detektie komunit v jazyku C. Naviac je knižnica paralelizovaná pomocou *plinq* pre maximalizáciu rýchlosťi. [26]

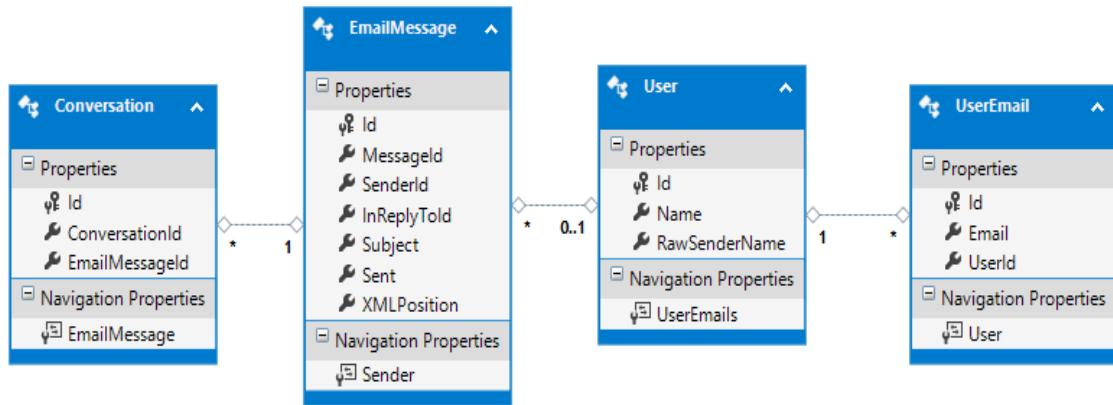
Canvas.js

Ďalšou použitou grafickou knižnicou je knižnica *canvas.js*. Je to HTML5 knižnica, ktorá umožňuje responzívne vykreslovanie grafov. Táto knižnica bola použitá pre vykreslenie koláčového a stĺpcového grafu. [27]

7.5 Import dát

Do aplikácie je možné nahrať XML súbor, ktorý je spracovaný uloženou SQL procedúrou, ktorá rozparsuje emailové dáta na jednodlivé entity - *User*, *EmailMessage*, *UserEmail* a *Conversation*

a uloží ich do SQL databázy. S tým uloženými emailovými dátami používam pre import do aplikácie *Entity framework*, ktorý zaručuje prenos dát medzi aplikáciou a SQL databázou.



Obr. 26: Doménový model

7.6 Implementácia

Aplikácia je napísaná v jazyku C#, grafické rozhranie je naimplementované pomocou návrhového vzoru Model View Controller a graf bol vizualizovaný pomocou knižnice vis.js. Aplikácia bola vyvíjaná vo Visual Studiu 2017.

7.6.1 Metóda pre získanie emailových dát

Pre získanie emailových dát z emailovej schránky som naimplementovala metódu, ktorá sa pomocou protokolu IMAP pripojí na danú emailovú schránku a stiahne emaily vo forme XML súboru.

EMAIL SERVER CONFIGURATION

Email		
veronika.uhrova@globallogic.com		
Username	veronika.uhrova@globallogic.com	
Password	*****	
Server address	Port	
imap.gmail.com		993
<input checked="" type="checkbox"/> Use secure connection(SSL)		
<input type="button" value="Submit"/>		

Obr. 27: Príklad konfigurácie emailu pre získanie emailov

7.6.2 Konštrukcia siete

Rozdiel medzi príspom rôznych štúdií a mojím prístupom pri konštrukcii grafu z emailového datasetu je v konštrukcii komunikačnej siete. Ako základnú stavebnú jednotku siete som si zvolila **konverzáciu**. Inšpirovala som sa prácou autorov Kudělka, Horák, Zehnaloová [7]. Konverzácia je teda súbor emailov, ktorá začína jediným emailom, obsahuje najmenej 2 emaily a dvoch rôznych odosielateľov. Vrcholom siete (grafu) sa teda stane užívateľ, ktorý bol ako odosielateľ aspoň v jednej takejto konverzáции. Hrana medzi užívateľmi je zostrojená medzi užívateľmi, ktorí boli spolu v jednej konverzáции ako odosielatelia. Takto teda vyváram neorientovaný nevážený graf. Pre konverzáciu ešte ukladám čas jej začiatku, užívateľ si následne v aplikácii môže zvoliť časový rozsah konverzácií.

7.6.3 Triedy pre graf, vrcholy a hrany

Pre uloženie siete v pamäti slúži generická trieda `Graph<T>`. Vrcholy a hrany drží ako `Dictionary<int, HashSet<Node<T>>>`, čiže ako mapu vrcholov s ich susednými vrcholmi. Pre uloženie vrcholov a hrán grafu slúžia zoznamy hrán a vrcholov uložené ako `HashSet<Node<T> >` a `HashSet<Edge<T>>`. Trieda je generická preto, aby bolo možné vytvoriť graf pre rôzne entity. Triedy reprezentujúce vrcholy a hrany sú tiež generické a predstavujú ich `Node<T>` a `Edge<T>`. Pre identifikáciu komunity, bola vytvorená trieda `Community<T>`.

8 Experimenty

V tejto kapitole popisujem experimenty, ktoré som previedla nad rôznymi emailovými sadami vo vytvorennej aplikácii. Popisujem prípravu a import dát, ďalej vizualizáciu jednotlivých datasetov a tiež výsledky a porovnanie vybraných analytických metód. V prvej časti popisujem analýzu agregovanej sady ako celkovú analýzu emailovej komunikácie tímu a v druhej časti popisujem analýzu emailovej komunikácie jednotlivca.

Poznámka: Z dôvodu ochrany osobných údajov sú všetky mená osôb a domén anonymizované.

8.1 Analýza emailovej komunikácie tímu

Analýzu emailovej komunikácie tímu som previedla na agregovanej emailovej sade, ktorá obsahuje emaily od štyroch jednotlivcov. Základné informácie o tomto datasete sú uvedené v tabuľke.

	počet
Emaily	21738
Konverzácie	4149
Požívateľia	370
Používateľia, ktorí sú aspoň v jednej konverzáции	273
Emaily poslané z pracovného emailu	20367

Tabuľka 1: Základné informácie o datasete

8.1.1 Príprava a import dát

Ako už bolo spomenuté, pre samotnú analýzu som nepoužila žiadny verejne dostupný dataset, ale použila som svoju emailovú sadu a emailové sady od mojich troch kolegov z práce. V dobe, keď som od nich emaily požadovala, moja aplikácia ešte nebola hotová a tak sa pre získanie ich emailov použila externá aplikácia *TeamNet Data* [28]. Emailové sady boli poskytnuté v súbore formátu XML.

Kedže sme kolegovia a pracujeme spolu v tíme, nachádzali sme sa viac krát v rovnakej emailovej retazi, takže sa v sadách vyskytovali emaily duplikátne. Pre spracúvanie emailov som vytvorila SQL procedúru, ktorá jednotlivé XML súbor načíta, rozdelí emaily, používateľov a konverzácie na jednolivé entity a nainštalauje ich do SQL databázy. Pre odstránenie duplikátov som použila SQL skript, ktorý zaručil odstránenie duplikátnych položiek.

Po spustení aplikácie mám na výber stiahnuť si svoj XML súbor o svojej emailovej schránky, nainštalať ho do aplikácie a ďalej preraziať svoje emailové dátá. Kedže emailové sady som už mala predpripravené, mohla som postupovať priamo k vizualizácii a analýze.

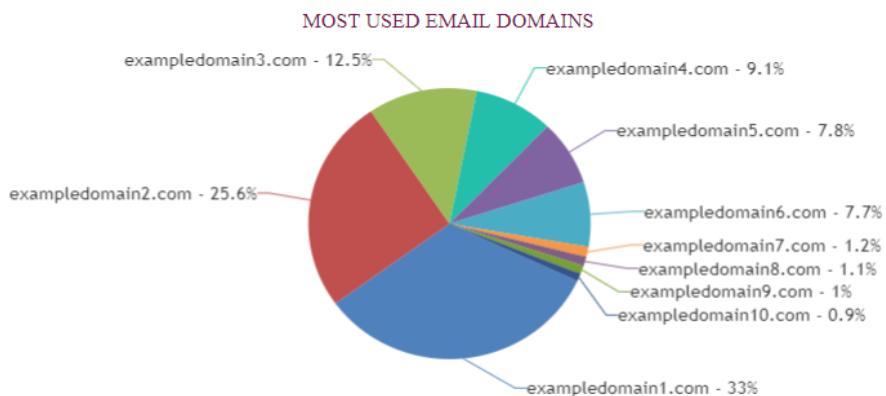
8.1.2 Vizualizácia datasetu

Aplikácia umožňuje výber vizualizácie jednotlivých členov tímu a tiež celého tímu celkovo. S výberom daného datasetu sa pre danú sieť zobrazia aj základné informácie ako počet emailov, odosielateľov emailov, najpoužívanejšie emailové domény a podobne.



Obr. 28: Základné informácie o tímovej sieti.

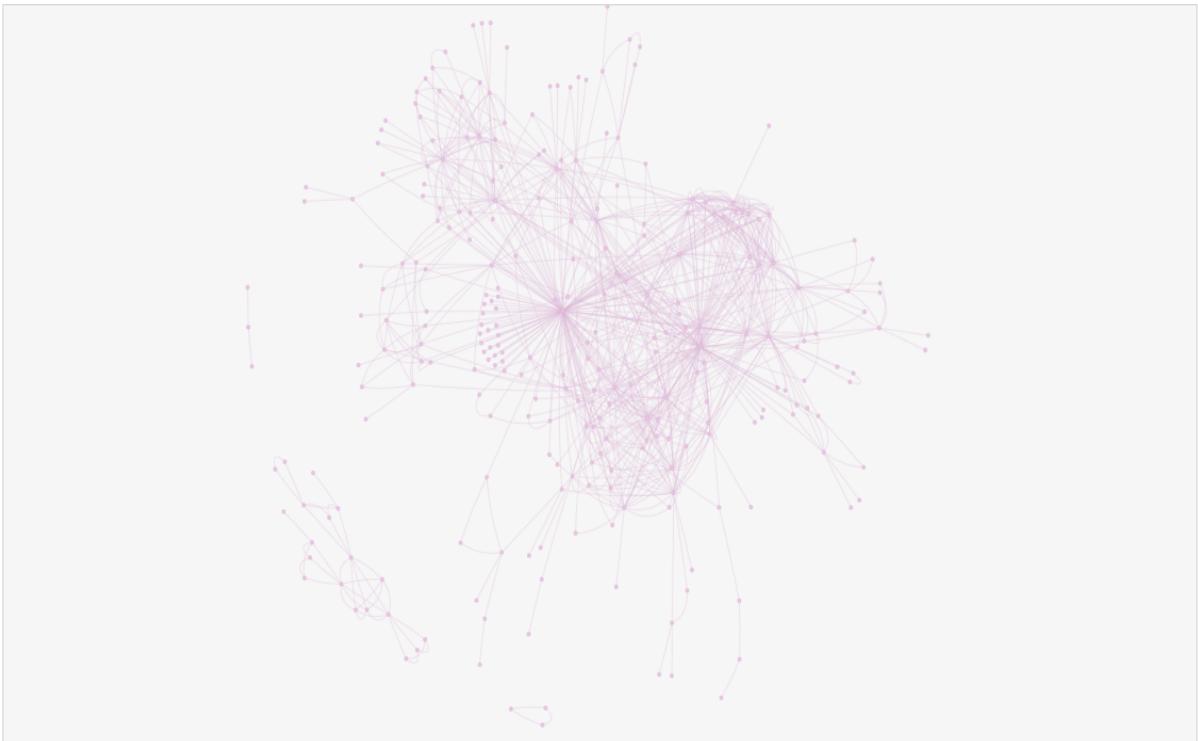
Na obr. 28 sú zobrazené základné informácie o agregovanej sieti. Štýria sme celkovo napísali 21738 emailov z toho bolo detektovaných 4149 konverzácií. Celkovo sa na emailovej komunikácii podieľalo 370 používateľov. Najviac emailov v agregovanej sade poslal používateľ *User 328*. Hodina, kedy sa posielalo celkovo najviac emailov bola medzi ôsmou a deviatou hodinou ráno, čiže to je čas, kedy ľudia prídu do práce a prvé, čo urobia je, že si skontrolujú emaily.



Obr. 29: Najviac používané emailové domény.

Na obr. 29 sú zobrazené emailové domény, z ktorých používatelia najviac posielali emaily. Najviac emailov sa poslalo z domén, ktoré sú oficiálne domény spoločnosti, v ktorej pracujem. Ostatné domény patria zákazníkom, s ktorými ako spoločnosť spolupracujeme.

Na obr. 30 je vizualizácia celkovej siete tímu v základnej podobe.



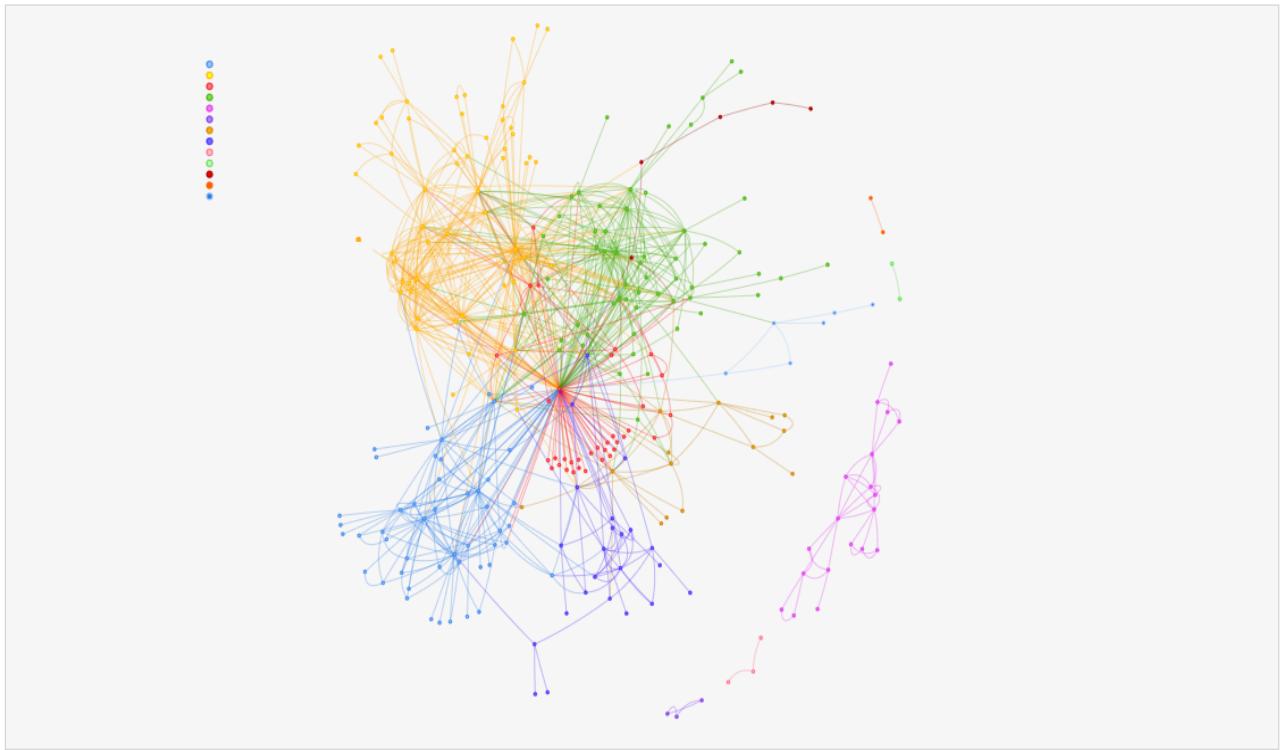
Obr. 30: Vizualizácia siete.

8.1.3 Detekcia komunit

Aplikácia umožňuje detektovať komunity v sieti a následne ich vizualizovať. Pre ďalšiu analýzu poskytujem v nasledujúcej tabuľke základné informácie o členoch tímu. Pre analýzu komunít využívam to, že každý nastúpil do práce v iný rok, takže na prelome týchto rokov by mal byť zaznamenaný nárast komunít.

Meno	Aktuálna pozícia	Dátum nástupu do firmy
Člen tímu 1	Test engineer	1.7.2011
Člen tímu 2	Junior Developer	1.9.2016
Člen tímu 3	Medior Developer	1.6.2017
Člen tímu 4	Lead Developer	1.11.2010

Tabuľka 2: Informácie o členoch tímu



Obr. 31: Vizualizácia komunít v tímovej sieti za celkový čas

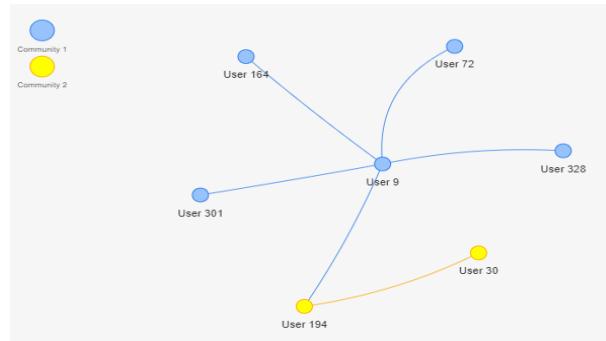
Celkovo bolo v sieti detekovaných 13 komunít. Najväčšia komunita má 57 uzlov, najmenšia má 2 uzly. Celkové zloženie komunít je zobrazené na nasledujúcom obrázku.



Obr. 32: Rozloženie komunít v tímovej sieti za celkový čas

8.1.3.1 Zmeny komunít v čase

Zmeny komunít som zaznamenávala v dátumoch, kedy jednotliví členovia tímu nastupovali do práce. Prvým časovým úsekom, ktorý som sledovala bolo obdobie medzi 1.11.2010 - 1.7.2011, kedy bol zamestnancom spoločnosti zatiaľ len jeden z nás. Boli detekované dve komunity, rozloženie uzlov je zobrazené na obrázku 33



Obr. 33: Rozloženie komunít za prvý časový úsek

Ďalší interval, ktorý som zvolila bol interval medzi 1.11.2010 - 31.8.2016, ktorý reprezentuje čas, kedy boli zamestnancami dva jednotlivci. Detekovaných bolo 9 komunít.



Obr. 34: Rozloženie komunít za druhý časový úsek

Do príchodu ďalšieho kolegu (interval 1.11.2010 - 31.5.2017) bolo detekovaných 12 komunít, čiže mojim príchodom do firmy pribudli tri komunity, to znamená že príchodom ďalšieho kolegu počet komunít vzrástol o jednu komunitu. Na ďalšom obrázku je zobrazené rozloženie komunít v treťom intervale.

COMMUNITIES	
Community 1	46 nodes
Community 2	44 nodes
Community 3	27 nodes
Community 4	27 nodes
Community 5	21 nodes
Community 6	19 nodes
Community 7	3 nodes
Community 8	24 nodes
Community 9	3 nodes
Community 10	2 nodes
Community 11	2 nodes
Community 12	6 nodes

Obr. 35: Rozloženie komunít za tretí časový úsek

8.1.4 Ego siet

Pre analýzu ego siete som vybrala troch jednotlivcov, z ktorých som vytvorila ego uzol a vytvorila pre nich ego siet. Porovnávala som jednotlivé metriky a počet komunít, ktoré spájali. Vyberala som uzly s rozdielnou centralitou, aby som mohla sledovať porovnanie medzi výsledkami. Porovnanie je možné vidieť v tabuľke.

	Stupeň uzla	Počet prepojených komunít	Efektívna veľkosť	E-I index
Aktér 1	129	13	127	0.52
Aktér 2	81	11	72	0.06
Aktér 3	20	11	18	0.4

Tabuľka 3: Informácie o vytvorennej ego sieti

Podľa veľkosti, čiže stupňa uzla môžem povedať, že uzol s veľkosťou 129 bude mať určie väčší sociálnu podporu od ostatných aktérov v sieti, väčší prísun k zdrojom a informáciám v porovnaní s uzlami s veľkosťou 81 alebo 20. Čo sa týka počtu prepojených komunít sú ale títo aktéri na tom podobne a spájajú porovnatelné množstvo komunít. Podľa E-I indexu môžem povedať, že ego siet každého aktéra je heterofílna a tak sa v svojej práci stretávajú so širokou skálou ľudí, nielen s ľuďmi z podobného okolia.

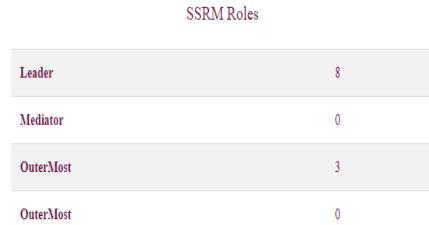
8.1.5 Analýza rolí

Analýzu rolí som prevádzala na sieti s vytvoreným ego uzlom (ako ego uzol som zvolila uzol s najväčším stupňom).

8.1.5.1 SSRM

V na obr. 36 sú zobrazené počty detekovaných štrukturálnych rôl v sieti. Uzly, ktoré boli de-

tekované ako *leader* zastávajú v reálnom živote pozíciu *Lead developer*, *Tester*, *Product owner*, *HR manager*, *Project manager*, *Sales manager*, *IT Consultant* a ďalší *Project manager*. Žiadny uzol neboli detekovaný ako *Mediator*. Tri uzly boli detekované ako *Outermost*, čo majú byť menej dôležitý jedinci a naozaj som sa s nimi ani s ich menom v práci nestretla.



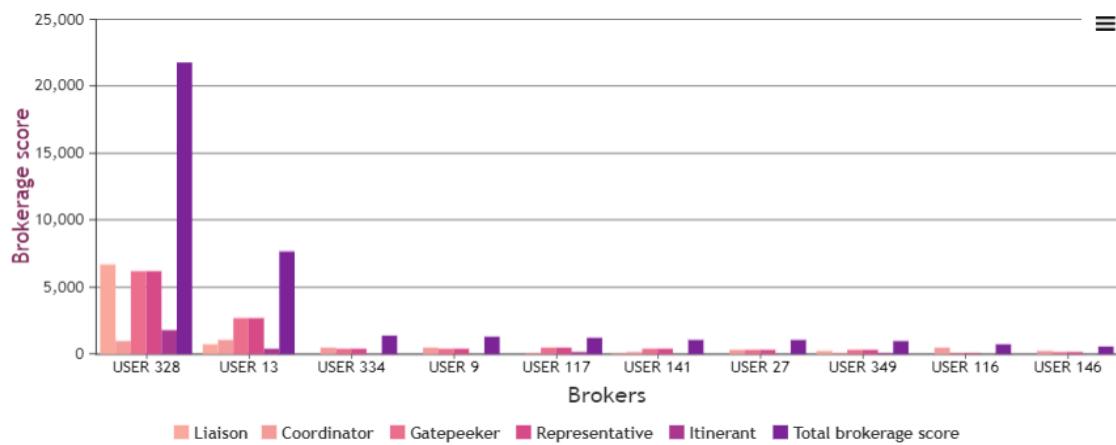
Obr. 36: Počet detekovaných štrukturálnych rôl

8.1.5.2 Brokerage

V na obr. 37 a 39 je zobrazených desať najväčších *broker* aktérov v sieti spolu s grafom ich čiatkovým skóre pre každú *brokerage* rolu, ako aj celkové *brokerage* skóre.

TOP 10 BROKERS							
	Name	Coordinator	Itinerant	Gatekeeper	Representative	Liaison	Total
1	USER 328	976	1758	6166	6166	6646	21712
2	USER 13	1084	404	2708	2708	744	7648
3	USER 334	498	12	420	420	26	1376
4	USER 9	460	14	408	408	40	1330
5	USER 117	116	130	480	480	0	1206
6	USER 141	162	20	416	416	72	1086
7	USER 27	344	6	326	326	40	1042
8	USER 349	44	54	324	324	210	956
9	USER 116	506	0	110	110	4	730
10	USER 146	204	6	176	176	20	582

Obr. 37: Desať aktérov s najväčším *brokerage* skórom



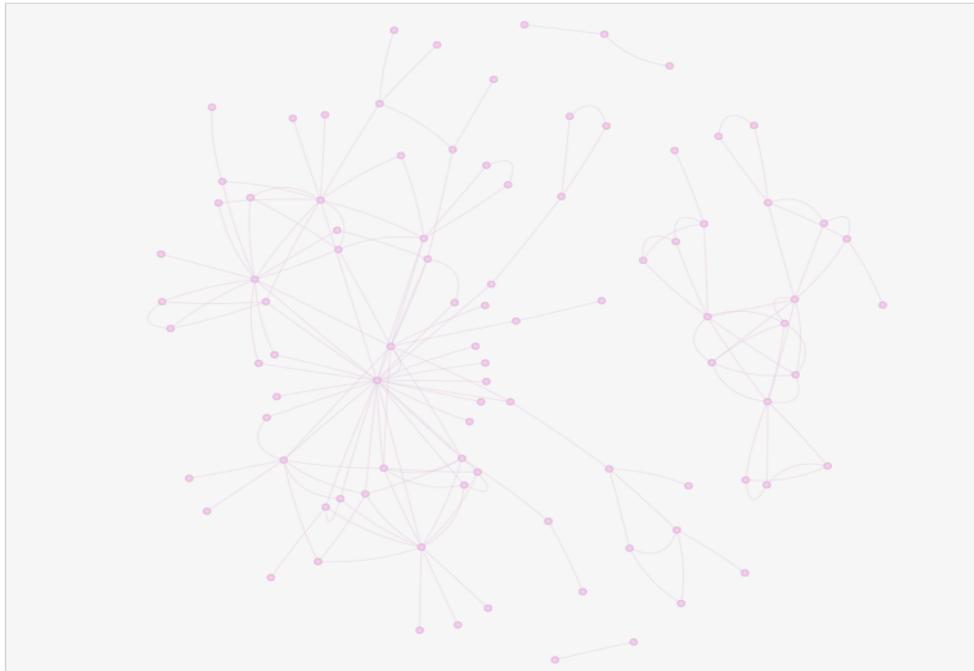
Obr. 38: Desat aktérov s najväčším *brokerage* skórom - graf

8.2 Analýza jednotlivca

V tejto časti popisujem analýzu jednotlivca od získania emailových dát z emailového účtu, importu do aplikácie a zobrazenie výsledkov metód v aplikácii. Pre analýzu jednotlivca používam svoju emailovú sadu pre demonštrovanie získavania a importu emailov.

8.2.1 Príprava a import dát

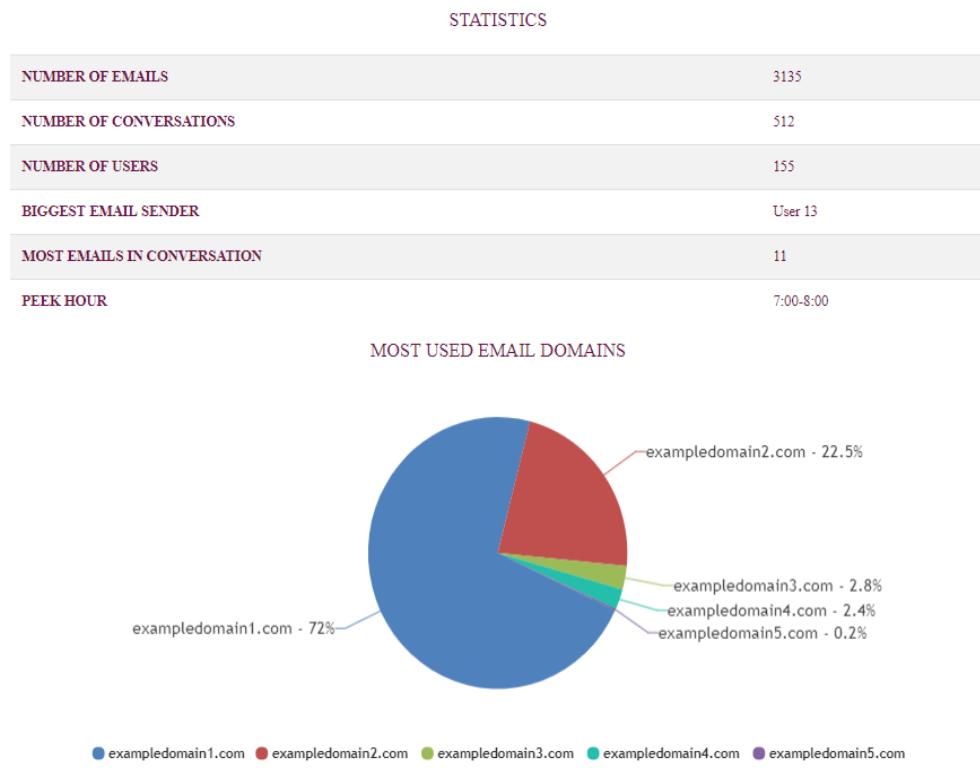
Pre získanie emailových dát z emailovej schránky používam navrhnutú aplikáciu. Po zadaní emailového účtu, používateľského mena, hesla, adresu servera a portu získam XML súbor s emailami, ktorý naimportujem do aplikácie a zobrazí sa mi moja emailová sieť.



Obr. 39: Analýza jednotlivca - základná vizualizácia

8.2.2 Informácie o datasete

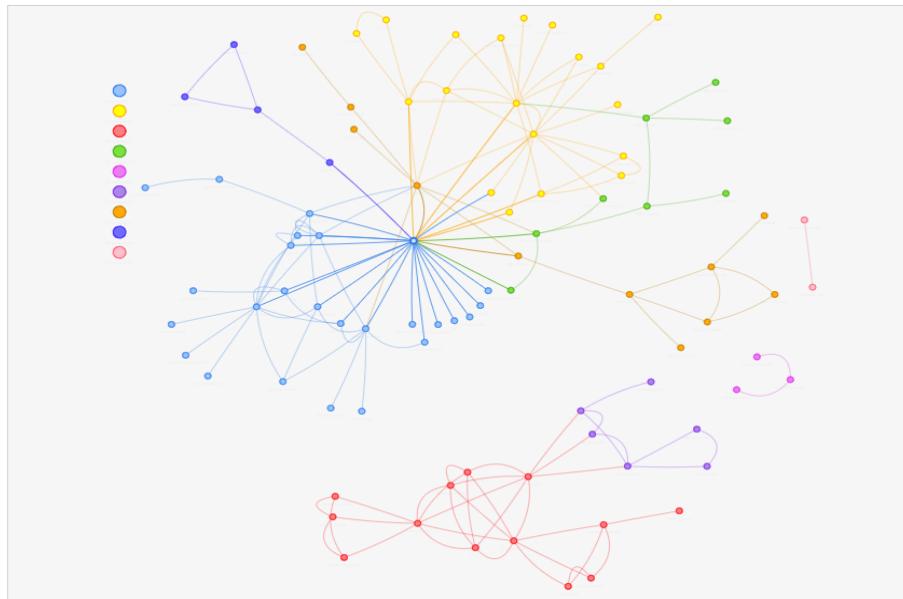
Spolu s vizualizáciou datasetu sa zobrazia aj základné štatistiky o datasete. Moja emailová schránka obsahuje 3135 emailov, ktoré napísalo 155 používateľov a bolo detekovaných 512 konverzácií. Osoba, ktorá napísala najviac emailov bol používateľ, ktorý zastáva pozíciu *Project manager*. Hodina, kedy sa písalo najviac emailov bola medzi siedmou a ôsmou hodinou ráno. Celkový sumár spolu s najpoužívanejšími emailovými doménami je na obrázku 41.



Obr. 40: Analýza jednotlivca - základné štatistiky

8.2.3 Detekcia komunít

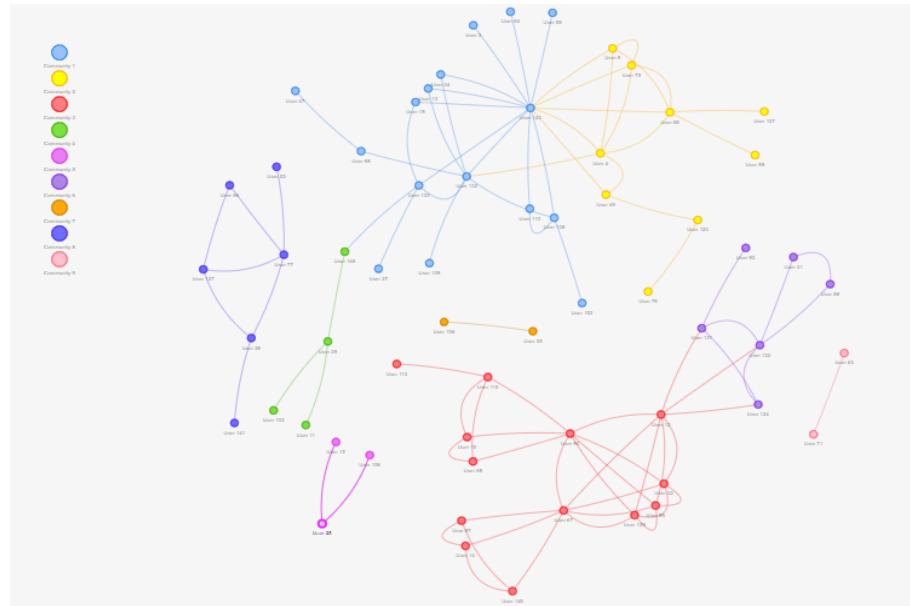
Celkovo bolo v dataste detektovaných 9 komunít. Najväčšia komunita obsahuje 26 uzlov, najmenšia komunita obsahuje 2 uzly.



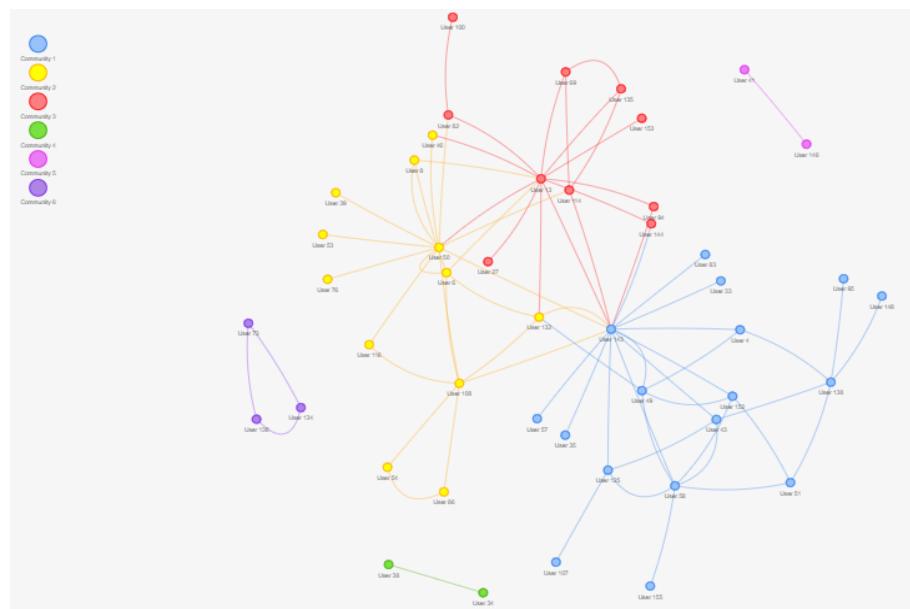
Obr. 41: Analýza jednotlivca - vizualizácia komunít

8.2.3.1 Zmeny komunít v čase

Pre zmeny komunít v čase som využila obdobia, kedy som pracovala na iných projektoch a teda sa komunity môžu v týchto intervaloch lísiť. Obdobie prvého projektu je v intervale od 1.9.2016 do 1.3.2017 a obdobie druhého je od 1.3.2017 do súčasnej doby, povedzme do 31.3.2018.



Obr. 42: Analýza jednotlivca - Vizualizácia komunít v prvom časovom intervale



Obr. 43: Analýza jednotlivca - vizualizácia komunít v druhom časovom intervale

Ako vidieť na obrázkoch 42 a 43 zloženie komunít sa lísi, či už zložením jednotlivých uzlov, tak aj velkosťou a počtom, čo je samozrejme prirodzené, keďže som komunikovala v týchto

intervaloch s inými osobami. V prvom intervale bolo detekovaných 9 komunití, najväčšia komunita mala 16 uzlov, najmenšia 2 uzly. Čo sa týka druhého intervalu, bolo detekovaných 6 komunití, najväčšia komunita obsahovala 17 uzlov najmenšia rovnako 2 uzly.

8.2.4 Ego sieť

Pre analýzu ega som vybrala cielene troch jednotlivcov zo siete, ktorí pracujú na rozdielnej pozícii. Vybrala som jedného developera, jedného projektového manažéra a testera. Z každého z nich vytváram v sieti ego a v nasledujúcej tabuľke porovnávam ich silu v sieti.

Aktér	Pozícia	Prepojené komunity	Efektívna veľkosť	E-I index
Aktér 1	developer	8	28	-0.06
Aktér 2	projektový manažér	9	15	-0.25
Aktér 3	tester	4	5	0

Tabuľka 4: Informácie o vytvorenej ego sieti

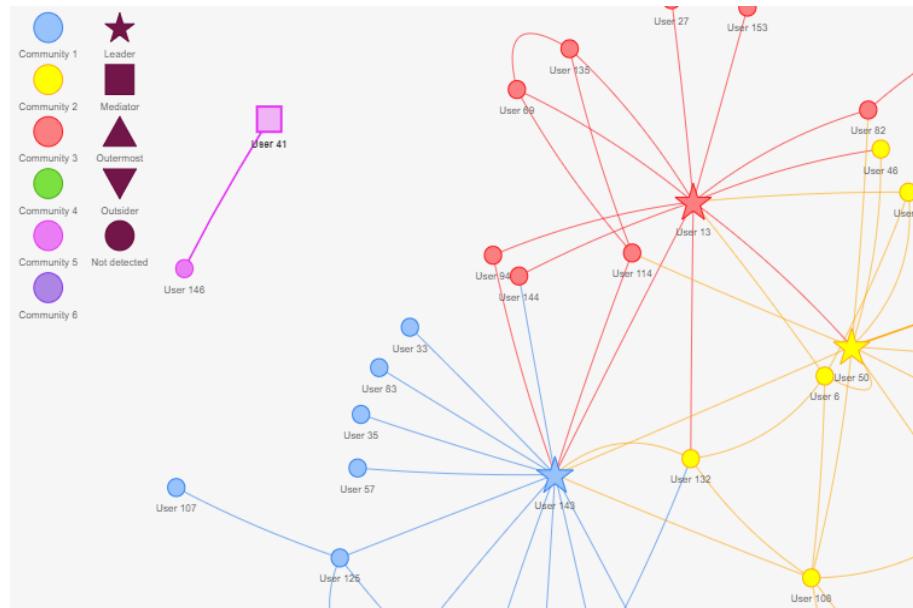
Podľa pozície by sme mohli predpokladať, či daný aktér spája viac alebo menej komunití, pretože projektový manažér sa určite vo svojej práci stretáva s rôznorodejšími osobami ako tester alebo developer. To ukázala aj analýza ega, kedy projektový manažér ako ego spája všetkých 9 komunití, kým tester len 4 komunity. Tiež sa líši efektívna veľkosť, ktorá udáva počet neredundantných uzlov. Kým developer má 28 nerundantných uzlov, projektový manažér má len 15, čo môže znamenať, že v reálnom svete projektový manažér môže čerpať informácie cez rôzne iné uzly, ktorí developer sa sústredí na najbližšie spojky. E-I index udáva, že ego siete sú homofílné a teda aktéri v tomto datasete komunikovali výlučne s ľuďmi v ich komunite.

8.2.5 Analýza rolí

Analýzu rolí som prevádzala na sieti s detekovaným ego uzlom, ktorý som zvolila podľa najväčšieho stupňa uzla.

8.2.5.1 SSRM

Analýza štrukturálnych rolí dopadla podľa očakávaní. Boli detekované tri uzly s rolou *leader* a jeden uzol s rolou *mediator*. Uzly detekované ako *leader* pracujú na pozíciach *Project Manager*, *Product owner* a *Developer*. Možno trochu nezvyčajné je, že mňa ako developera môj algoritmus identifikoval ako *leader* rolu. Keď to premietnem do reálneho života, kde je v tíme jediná žena, môžem povedať, že táto detekcia dáva zmysel.



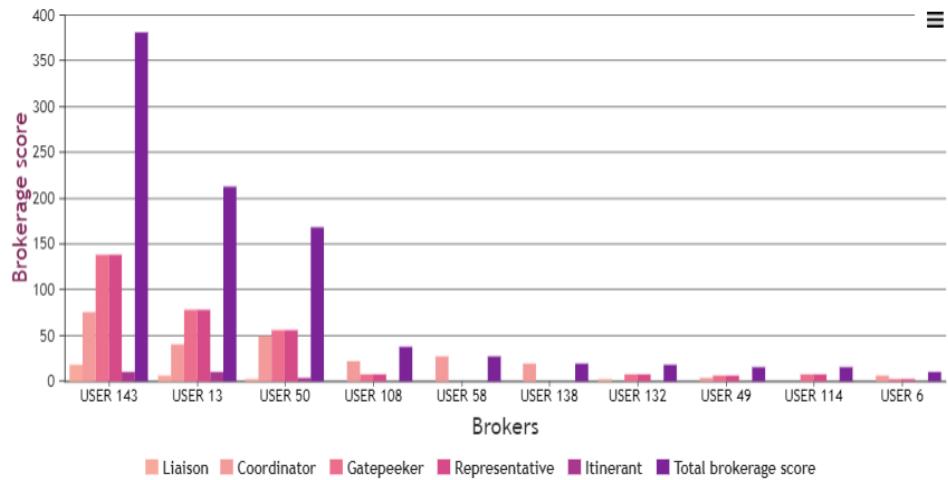
Obr. 44: Analýza jednotlivca - detail detekovaných SSRM rolí

8.2.5.2 Brokerage

V na obr. 45 a 46 je zobrazených desať najväčších *broker* aktérov v sieti spolu s grafom ich čiatkovým skórom pre každú *brokerage* rolu, ako aj celkové *brokerage* skóre.

TOP 10 BROKERS							
	Name	Coordinator	Itinerant	Gatekeeper	Representative	Liaison	Total
1	USER 143	76	10	138	138	18	380
2	USER 13	40	10	78	78	6	212
3	USER 50	50	4	56	56	2	168
4	USER 108	22	0	8	8	0	38
5	USER 58	28	0	0	0	0	28
6	USER 138	20	0	0	0	0	20
7	USER 132	0	0	8	8	2	18
8	USER 49	4	0	6	6	0	16
9	USER 114	0	0	8	8	0	16
10	USER 6	6	0	2	2	0	10

Obr. 45: Desať aktérov s najväčším *brokerage* skórom



Obr. 46: Desať aktérov s najväčším *brokerage* skórom - graf

Používatelia, ktorí boli detekovaní ako *broker* jednotlivci, zastávajú vo firme pozície ako *Asistent, Lead developer, Product owner, Consultant alebo Marketing specialist.*

9 Záver

Práca splnila všetky zadané ciele. Boli naštudované rešerše obdobných riešení a analytických prístupov a vybrané metódy analýzy sietí vhodné pre analýzu emailovej komunikácie. Na tomto základe bolo navrhnuté používateľské rozhranie pre analýzu emailovej komunikácie a vizualizáciu analytických výstupov. Toto používateľské rozhranie umožňuje získavanie emailových správ z vybraného zdroja, import týchto alebo inak získaných dát, vizualizáciu a zobrazenie výsledkov jednotlivých metód pre analýzu emailovej komunikácie. Aplikácia umožňuje zobrazenie základných informácií o emailovej komunikácii, vizualizáciu siete, detekciu komunit, tvorbu ego siete, detekciu štrukturálnych rolí a detekciu *brokerage* rolí v emailovej sieti. Následne boli uvedené experimenty analýzy emailovej komunikácie tímu a emailovej komunikácie jednotlivca.

9.1 Možnosti rozšírenia a zdokonalenia práce

9.1.1 Možné rozšírenia aplikácie

Vizualizácia veľkých grafov

Využiteľnosti aplikácie by pomohlo, keď vedela vizualizovať aj rádovo väčšie grafy. So zobrazovaním rozsiahlych grafov sa v práci nepočítalo a už pri zobrazovaní emailov celého tímu nebolo celkom pole pre vykreslenie grafu dostačujúce, čo bolo limitované aj knižnicou *vis.js*.

Viac možností filtrovania

V aplikácii môže používateľ filtrovať emailovú komunikáciu podľa času. Mohlo by byť užitočné nastavenie filtru aj podľa iného kritéria.

Zobrazenie viac informácií o hranách a vrcholoch

Aplikácia umožňuje zobraziť informácie o stupni vrcholu. Pri analýze grafu ako takom by bolo dobré rozšíriť tieto informácie napríklad o metriky a ďalšie charakteristiky siete.

Rozšírenie analytických metód

Aplikácia sa sústredí na analýzu siete zo štrukturálneho hľadiska, každopádne by bolo užitočné aplikáciu obohatiť o ďalšie analytické metódy, ktoré by pomohli emailovú komunikáciu spoznať ešte hlbšie.

Literatura

- [1] “Steve borgatti: Brokerage.” <http://www.analytictech.com/Essex/Lectures/Brokerage.pdf>.
- [2] J. Diesner, T. L. Frantz, and K. M. Carley, “Communication networks from the enron email corpus “it’s always about the people. enron is no different”,” *Computational & Mathematical Organization Theory*, vol. 11, no. 3, pp. 201–228, 2005.
- [3] X. Fu, S.-H. Hong, N. S. Nikolov, X. Shen, Y. Wu, and K. Xuk, “Visualization and analysis of email networks,” in *Visualization, 2007. APVIS’07. 2007 6th International Asia-Pacific Symposium on*, pp. 1–8, IEEE, 2007.
- [4] A. Chapanond, M. S. Krishnamoorthy, and B. Yener, “Graph theoretic and spectral analysis of enron email data,” *Computational & Mathematical Organization Theory*, vol. 11, no. 3, pp. 265–281, 2005.
- [5] G. Tang, J. Pei, and W.-S. Luk, “Email mining: tasks, common techniques, and tools,” *Knowledge and Information Systems*, vol. 41, no. 1, pp. 1–31, 2014.
- [6] A. Abnar, M. Takaffoli, R. Rabbany, and O. R. Zaïane, “Ssrm: structural social role mining for dynamic social networks,” *Social Network Analysis and Mining*, vol. 5, no. 1, p. 56, 2015.
- [7] S. Zehnalova, Z. Horak, and M. Kudelka, “Email conversation network analysis: Work groups and teams in organizations,” in *Advances in Social Networks Analysis and Mining (ASONAM), 2015 IEEE/ACM International Conference on*, pp. 1262–1268, IEEE, 2015.
- [8] “Do millennial and gen z consumers still use email?” <https://www.bluecore.com/blog/do-millennials-use-email/>. Navštívené: 2017-03-30.
- [9] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.
- [10] M. E. Newman, “The structure and function of complex networks,” *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003.
- [11] A.-L. Barabási, *Network science*. Cambridge university press, 2016.
- [12] R. D. Luce and A. D. Perry, “A method of matrix analysis of group structure,” *Psychometrika*, vol. 14, no. 2, pp. 95–116, 1949.
- [13] G. W. Flake, S. Lawrence, and C. L. Giles, “Efficient identification of web communities,” in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 150–160, ACM, 2000.

- [14] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, “Defining and identifying communities in networks,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 9, pp. 2658–2663, 2004.
- [15] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.
- [16] N. Crossley, E. Bellotti, G. Edwards, M. G. Everett, J. Koskinen, and M. Tranmer, *Social network analysis for ego-nets: Social network analysis for actor-centred networks*. Sage, 2015.
- [17] R. V. Gould and R. M. Fernandez, “Structures of mediation: A formal approach to brokerage in transaction networks,” *Sociological methodology*, pp. 89–126, 1989.
- [18] P. V. Marsden, “Brokerage behavior in restricted exchange networks,” *Social structure and network analysis*, vol. 7, no. 4, pp. 341–410, 1982.
- [19] R. S. Burt, *Structural holes: The social structure of competition*. Harvard university press, 2009.
- [20] K. Stovel and L. Shaw, “Brokerage,” *Annual Review of Sociology*, vol. 38, pp. 139–158, 2012.
- [21] E. S. Spiro, R. M. Acton, and C. T. Butts, “Extended structures of mediation: Re-examining brokerage in dynamic networks,” *Social Networks*, vol. 35, no. 1, pp. 130–143, 2013.
- [22] R. DeJordy and D. Halgin, “Introduction to ego network analysis,” *Boston MA: Boston College and the Winston Center for Leadership & Ethics*, 2008.
- [23] R. S. Burt, “Structural holes and good ideas,” *American journal of sociology*, vol. 110, no. 2, pp. 349–399, 2004.
- [24] S. P. Borgatti, “Structural holes: Unpacking burt’s redundancy measures,” *Connections*, vol. 20, no. 1, pp. 35–38, 1997.
- [25] “Active Record vs. Repository pattern repository pattern.” <https://www.rarous.net/weblog/271-active-record-vs-repository-pattern.aspx>. Navštívené: 2018-01-15.
- [26] “Louvainsharp - fast louvain method of community detection in c#.” <http://www.markusmobius.org/software/louvainsharp-fast-louvain-method-community-detection-c>. Navštívené: 2017-12-12.
- [27] “Canvas.js.” <https://canvasjs.com/>. Navštívené: 2018-04-20.
- [28] “TeamNET data.” <http://inflex.cz:8075/TeamNETdata>. Navštívené: 2017-09-30.

10 Prílohy

Súčasťou diplomovej práce je CD, ktorého súčasťou je:

1. zdrojový kód aplikácie
2. použité datasety
3. SQL skripty