Univerzita Hradec Králové

Fakulta informatiky a managementu

Katedra informatiky a kvantitativních metod

Míry centrality v sociálních sítích

Diplomová práce

Autor: Jiří Mikeš

Studijní obor: Aplikovaná informatika, AI2-P

Vedoucí práce: Mgr. Jiří Haviger, Ph.D.

Hradec Králové Listopad 2015

Prohlášení

Prohlašuji, že jsem diplomovou práci zpracoval samostatně a s použitím uvedené literatury.

V Hradci Králové dne 13. listopadu 2015 ……

Jiří Mikeš

Poděkování

Rád bych tímto poděkoval vedoucímu diplomové práce Mgr. Jiřímu Havigerovi Ph.D. za připomínky a rady, které mi poskytl při psaní této práce. Děkuji.

Anotace

Náplní této diplomové práce je analýza sociálních sítí. Čtenář je seznámen s komplexními sítěmi a s veličinami, které jsou používány při jejich analýze. Pozornost je věnována především posuzování míry centrality aktérů sítě. Centralita porovnává aktéry podle jejich vlivu v síti a určuje tak jejich důležitost. Hlavním cílem práce je podrobné seznámení s pěti základními metrikami pro určení míry centrality. Nechybí vysvětlení, jakým způsobem jsou tyto metriky vypočítávány a vzorce pro jejich výpočet jsou aplikovány v ukázkových příkladech. V rámci praktické části je popsán vývoj aplikace pojmenované Míry centrality v programovacím jazyce Java. Tato aplikace umožňuje uživateli zadat libovolný graf sítě a následně vypočítá hodnoty pěti základních metrik centrality aktérů této sítě. Implementované algoritmy, pomocí kterých je centralita vypočítána, jsou podrobně vysvětleny. V poslední části práce je aplikace použita k analýze kolaborační sítě oblíbených filmových herců. Tato diplomová práce může posloužit především jako studijní materiál pro každého, koho zajímají míry centrality sítí.

Annotation

Title: Measures of centrality in social networks

The scope of this diploma thesis is the analysis of social networks. The reader learns about complex networks and gain knowledge about measures, which are used in network analysis. Attention is dedicated mainly to the evaluation of network actor´s centrality. Centrality compares actors by their influence and importance in network. The main goal of thesis is to provide knowledge about five main centrality measures. It is explained how these measures are calculated and their formulas are applied in sample examples. There is described development of the application Measures of Centrality in practical part, which is written in programming language Java. This applications allow user to set arbitrary network graph and the application then provides results of five main centrality measures of network actors. Implemented algorithms for calculating measures are explained in detail. The last part of thesis contains analysis of collaboration network of favorite film actors. This diploma thesis can be used primarily as study material for everyone, who is interested in measures of network centrality.

Obsah

1. Úvod

Každý, kdo čte tuto práci, má okolo sebe různé druhy sítí a sám je také součástí sítí společenských vztahů několika druhů. Sítě mohou být tvořeny hmatatelnými objekty, jako je tomu u elektrických rozvodných sítí, Internetu, sítě dálnic nebo u neuronových sítí. Opakem sítí hmatatelných objektů jsou sítě definované v abstraktním prostoru. Mezi ně patří sítě společenských vztahů nebo kooperací. (1)

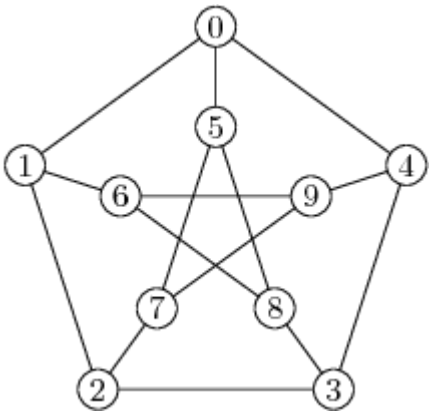
Studium sítí, z hlediska matematické teorie grafů, je jednou ze základních součástí diskrétní matematiky. Sítě jsou také často předmětem zájmu společenských věd a to typicky při vyhodnocování dotazníků, které zjišťují od respondentů, jaké mezi nimi probíhají interakce. Výsledky těchto dotazníků jsou poté použity k vytvoření modelové sítě, jejichž vrcholy znázorňují jednotlivé lidi a hrany interakci mezi nimi. Studium sociálních sítí se často věnuje posuzování centrality. Ta vyhodnocuje, které individuality jsou nejlépe propojené s ostatními nebo mají v síti největší vliv. (2)

V posledních letech došlo k podstatné změně ve výzkumu sítí. Pozornost se přesunula od analýzy malých grafů k rozsáhlejším systémům. Tato změna byla způsobena výrazným rozvojem a rozšířením informačních technologií, které nyní umožňují shromažďovat a analyzovat data ve velkém měřítku. S výrazným růstem sítí muselo dojít ke změně přístupu jejich analyzování. Otázky, které mohou být u malých sítí zodpovězeny snadno, jelikož lze nakreslit graf a z něho leccos vyčíst, je nyní nutné vyhodnocovat jiným způsobem. V počátcích oboru sloužilo lidské oko jako mocný analytický nástroj, který umožňoval z grafů malých sítí vyčíst její strukturu i další vlastnosti. V sítích složených z miliónů vrcholů není tento přístup možný. Ani s využitím moderních 3D vykreslovacích nástrojů nelze vytvořit graf rozsáhlé sítě, ve kterém by se člověk dokázal orientovat. Je tak nutné najít jiné cesty, kterými je možné charakterizovat strukturu a chování sítě. Nicméně i u rozsáhlých sítí je možné vytvořit některé přínosné modely. (2)

1.1. Základní reprezentace sítí

Existuje několik různých způsobů, kterými je možné reprezentovat sítě v závislosti na jejich tvaru a velikosti. V této kapitole bude věnována pozornost běžně používanému základnímu způsobu reprezentace, který bude v této práci dále využíván.

Základními prvky sítě je sada uzlů $V = \{1, \dots, v\}$, kterými je síť tvořena. Uzly bývají někdy také označovaný jako vrcholy, agenti, aktéři nebo hráči sítě. Označení se mění v závislosti na typu sítě. Uzly zastupují prvky, které jsou sledovány (lidé, firmy, země, webová stránka atd.). Na obrázku 1 je síť reprezentovaná jednoduchým grafem, ve kterém jsou vrcholy označeny číslicemi. (3)



Obrázek 1: Reprezentace sítě, převzato z https://ksp.mff.cuni.cz/tasks/26/cook1.html

Sítě mohou být znázorněny neorientovanými grafy, ve kterých mohou být každé dva uzly propojeny hranou E. V takovém případě jsou uzly sousední a hrana je incidentní s danými dvěma uzly. Stupeň uzlu je definován jako počet hran, které do uzlu vstupují/vystupují.

Neorientované grafy je možné použít při zkoumání mnoha sociálních nebo ekonomických vztahů, kterými jsou například partnerství, přátelství, aliance, atd. V případě jiných situací, kdy jeden uzel může být propojený s druhým, aniž by existovalo i opačné propojení, je nutné použít grafy orientované. Tato situace nastává například při modelování citací mezi autory. (3)

V orientovaných grafech má každá hrana určitý směr, který se obvykle znázorňuje šipkami. Orientovaný graf G je dvojice (V,E), kde E je podmnožina kartézského součinu V × V. Uspořádané dvojice $(x, y) \in E$ se nazývají orientované hrany. Říká se, že orientovaná hrana $e = (x, y)$ vychází z x a končí v y. (4)

Dalším pojmem, který je nutné objasnit, jsou cesty. V grafu $G = (V, E)$ se cestou označuje posloupnost $P = (v_0, e_1, v_1, \dots, e_n, v_n)$, pro kterou platí $e_i = \{v_{i-1}, v_i\}$ a navíc $v_i \neq v_j$ pro $i \neq j$. Je to tedy posloupnost vrcholů, pro kterou platí, že v grafu existuje hrana z daného vrcholu do jeho následníka. Žádné dva vrcholy (a tedy ani hrany) se přitom neopakují. Délka cesty je u neohodnoceného grafu počet hran dané cesty. (3)

Termínem kružnice (cyklus) se označuje posloupnost vrcholů a hran $(v_0, e_1, v_1, \dots, e_t, v_t = v_0)$, kde vrcholy $v_0, \dots, v_{t-1}$ jsou navzájem různé vrcholy grafu G a pro každé $i = 1,2, \dots, t$ je $e_i = \{v_{i-1}, v_i\} \in E(G)$. Kružnice je tedy cesta délky minimálně 3, která má první a poslední vrchol posloupnosti stejný. Graf, který jako podgraf obsahuje kružnici, se nazývá cyklický. V opačném případě jde o graf acyklický. (4)

1.2. Komplexní sítě

Sítě, jejichž struktura je nepravidelná, složitá a dynamicky se vyvíjí v čase, se označují jako komplexní sítě (complex networks). Mezi komplení sítě se řadí dopravní sítě, telefonní sítě, Internet, World Wide Web, sítě spolupracujících herců ve filmech, citační sítě nebo také různé biologické a medicínské systémy, mezi které patří neuronové a genetické sítě nebo metabolické sítě. (1)

U složitých sítí je vhodné zabývat se jejími strukturami, vlastnostmi, interakcí prvků, projevy emergence atd. Základem k tomuto zkoumání je teorie grafů a teorie pravděpodobnosti. Věda zkoumající sociální sítě se nazývá Analýza sociálních sítí (social network analysis). Společně s vědami zabývajícími se dalšími složitými sítěmi formuje základy Teorie sítí (network science). (5)

V rámci rozsáhlých analýz sítí z různých oborů bylo dosaženo mnoha nečekaných a zajímavých výsledků. Výzkum komplexních sítí začal pokusy, jak definovat koncepty a míry, kterými by bylo možné charakterizovat topologii sítí z reálného světa. Hlavními výsledky těchto zkoumání byla řada sjednocujících zásad a vlastností, které jsou společné pro většinu reálných sítí. (1)

Tyto poznatky umožnily značné vylepšení modelování sítí. Předchozí modely navrhované v matematické teorii grafů se ukázaly jako nepoužitelné pro reálné potřeby. Vědci museli vyvinout nové modely kopírující strukturální vlastnosti pozorované ve skutečných topologiích a schopné zachytit růst sítě. Struktura reálné sítě je výsledkem dlouhodobého vývoje ovlivněného silami, které je formují a různými způsoby zasahují do funkcionality systému. (1)

1.2.1. Topologie reálných sítí

Mnoho přírodních a technologických systémů je tvořeno velkým množstvím vysoce propojených dynamicky se vyvíjejících jednotek. Základním přístupem, jak zachytit

globální vlastnosti těchto systémů, je modelovat je pomocí grafu, jehož uzly reprezentují zmíněné dynamické jednotky a spojení uzlů představuje interakce mezi těmito jednotkami. Jedná se samozřejmě o výraznou aproximaci, jelikož interakce mezi dynamickými jednotkami, která obvykle závisí na čase, prostředí a mnoha dalších faktorech, je v tomto případě reprezentována pouze binárním číslem, jelikož existují pouze dvě možnosti a to, že spojení mezi uzly buď existuje, nebo nikoliv. Nicméně, v mnoha případech může i tato aproximace poskytnout jednoduchou, ale stále o mnohém vypovídající reprezentaci celého systému. (1)

S rostoucí dostupností obrovských databází a vývojem mocných a spolehlivých analytických nástrojů mohou být topologické vlastnosti sítí z reálného světa zkoumány neustále lépe a lépe. Tento rozvoj umožňuje studium topologie interakcí ve velkém množství systémů, mezi které patří komunikační, společenské nebo biologické systémy. Zásadním výsledkem těchto aktivit bylo objevení skutečnosti, že i přes výrazné rozdíly v těchto systémech jsou všechny tyto sítě charakterizovány podobnými topologickými vlastnostmi, jakými jsou například relativně nízké vzdálenosti cest, vysoké shlukové koeficienty, stupně korelace a přítomnost společenských struktur. Všechny tyto vlastnosti výrazně odlišují reálné sítě od náhodných grafů a vzbuzují zájem zkoumat jevy tvarující jejich topologii. (1)

Odborné práce zabývající se analýzou komplexních sítí ukázaly, že mnoho různých systémů projevuje společné vlastnosti topologie komplexních sítí, které se tak již dají předem předpovídat a systém není náhodný. Významným krokem ve snaze porozumět komplexním sítím bylo objevení způsobu rozložení stupňů vrcholů podle Poissonova rozdělení pravděpodobnosti. (6)

1.2.1.1. Fenomén „malého světa“

Zásadním přínosem pro studium topologie reálných sítí jsou Milgramovy experimenty objevující fenomén tzv. malého světa. V rámci těchto experimentů nebyly modelovány žádné skutečné sítě, nicméně sdělily mnohé o struktuře sítě. Stanley Milgram byl experimentální psycholog, který se rozhodl měřit vzdálenost cest dopisů poslaných mezi lidmi v USA. Vybral dva adresáty, kterými byli student z Massachusetts a bankéř z Bostonu, a požádal 160 náhodně vybraných osob z Kansasu a Nebrasky, aby se pokusily doručit dopis adresátovi. Pokud odesílatelé znali adresáty, mohli jim dopis poslat přímo, jinak se ji snažili poslat někomu, kdo by adresáta pravděpodobně mohl znát. Většina dopisů se ztratila, ale přibližně čtvrtina dorazila do cíle. Dopisy prošly

během cesty rukami průměrně 6 prostředníků. Tento experiment byl také zárodkem k objevení konceptu 6 kroků odloučenosti. (7)

1.2.2. Dělení komplexních sítí

Pomocí grafů je možné modelovat několik sítí z různých vědeckých oborů. V této části práce budou rozebrány rozdíly mezi různými typy komplexních sítí z reálného světa.

1.2.2.1. Sociální sítě

Sociální síť tvoří lidé nebo jejich skupiny, mezi kterými existují nějaké sledované vztahy nebo interakce. Těmito spojujícími prvky mohou být například přátelství mezi jednotlivými lidmi, podnikatelské vztahy mezi společnostmi nebo sňatky spojující rodiny. Zachycovány jsou situace z několika oborů, mezi které patří sociologie, aplikovaná antropologie nebo sociální psychologie. (1)

Sociální sítě se často setkávají s problémy nepřesnosti, subjektivity a malého počtu případů. Sběr dat obvykle probíhá oslovením účastníků přímo prostřednictvím dotazníků nebo rozhovorů. Tyto metody jsou pracné a časově náročné, takže značně omezují velikost sledované sítě. Získané výsledky jsou navíc často ovlivněny subjektivním pohledem účastníků, například za přátelství může každý respondent považovat něco mírně odlišného než respondent jiný. Ačkoliv je tedy vkládáno velké úsilí do odstranění těchto nepřesností, často se v těchto studiích objevují. (2)

Z těchto důvodů se mnoho výzkumníků uchyluje k jiným metodám analyzování sociálních sítí. Jedním bohatým a relativně spolehlivým zdrojem jsou kolaborační sítě. Často se jedná o partnerské sítě, ve kterých účastníci spolupracují ve skupinách určitého druhu a spojení mezi uzly jsou ustanovována pomocí členství ve skupinách. Typickým příkladem tohoto zdroje je databáze spolupracujících filmových herců. Tato databáze je důkladně spravována na webu www.imdb.com. V této síti jsou propojeni herci, kteří hráli alespoň v jednom filmu společně. (2)

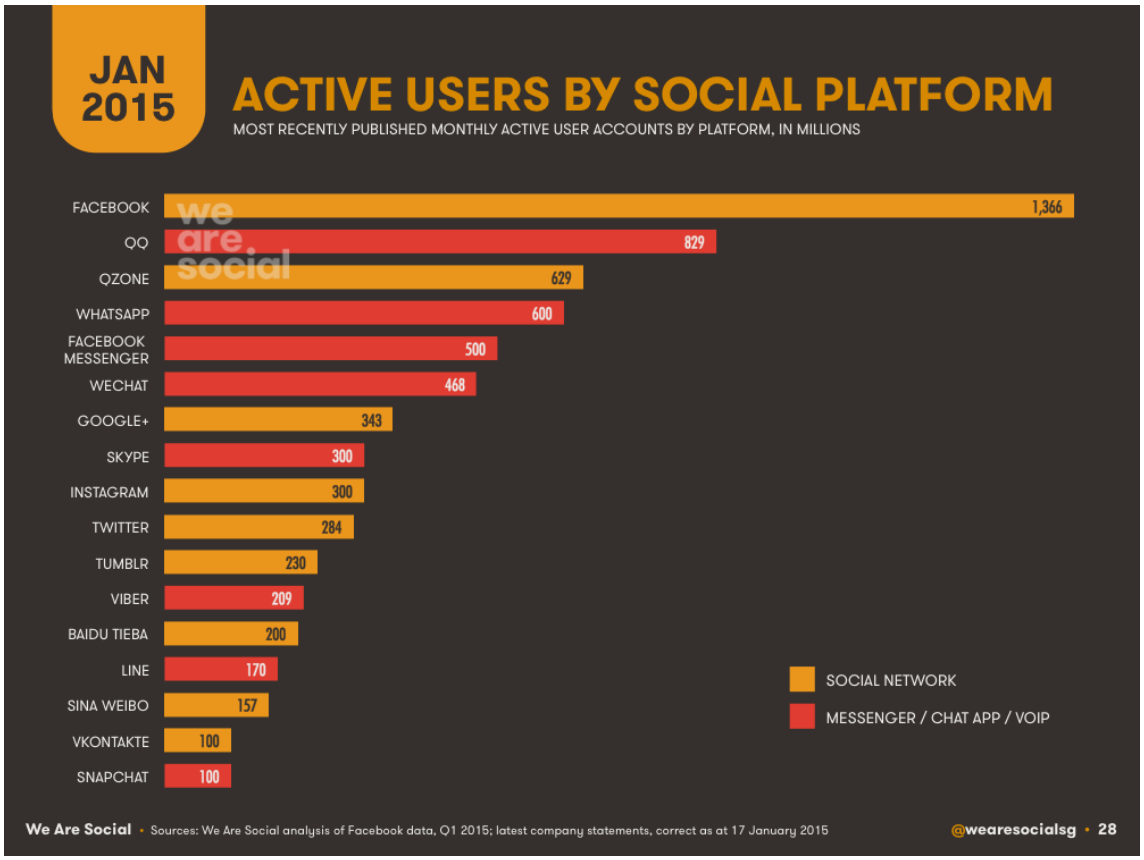
Zajímavým příkladem kolaboračních sítí je také síť vědeckých spoluprací. Díky těmto sítím došlo k lepšímu porozumění sociálních mechanismů panujících mezi publikujícími vědci. Tyto spoluautorské sítě jsou konstruovány tak, že jsou propojeni dva vědci (reprezentováni uzly) vždy, pokud se spolu podíleli alespoň na jednom díle. Tyto modely dokládají například existenci malého světa. Podobné problematice se věnují v další kapitole popsané citační sítě. (1)

Kolaborační sítě bývají někdy také znázorňovány pomocí bipartitních grafů, ve kterých jedna skupina vrcholů reprezentuje aktéry a druhá skupina díla. V takovém grafu lze spolupracující autory identifikovat podle toho, zda se setkávají alespoň v jednom vrcholu ze skupiny vrcholů reprezentujících díla. (8)

Pomocí modelů sítí jsou analyzovány například vztahy mezi dospívajícími delikventy nebo i mezi teroristickými buňkami. V rámci sociálních sítí došlo také k několika pokusům vyhodnotit sociální interakce mezi zvířaty. Další sítě se zabývaly zachycováním příslušnosti fotbalových hráčů, muzikantů nebo postav z různých literárních děl. (1)

Rozvoj komunikačních systémů, kterými jsou myšleny telefonní sítě a Internet, přinesl vznik nových forem společenských kontaktů, které vytvářejí zajímavou podskupinu sociálních sítí. Tyto virtuální kontakty také pomohly lépe pochopit, jak probíhá proces formování společenských interakcí. Studovány jsou obvykle interakce prováděné pomocí telefonu a emailu. Záznamy těchto konverzací tvoří další spolehlivý zdroj dat analýzy sociální sítí. (1)

Teorie sítí se také výrazně věnuje moderním online sociálním sítím, mezi které patří Facebook, Twitter, Instagram apod. Podle marketingového průzkumu překonal Facebook v průběhu roku 2010 v počtu návštěv Google a stal se tak nejnavštěvovanějším webem. Podobné popularitě se těší i další online sociální sítě, proto je nepochybně vhodné věnovat se také jejich analýze. (9)



Obrázek 2: Počty aktivních uživatelů online sociálních sítí a chatovacích aplikací, převzato z: http://wearesocial.sg/blog/2015/01/digital-social-mobile-2015/

V případě online sociálních sítí se obvykle v rámci analýzy vytváří graf, jehož vrcholy reprezentují uživatele a hrany znázorňují přátelství mezi nimi. V grafech jednotlivých online sociálních sítí se mohou projevovat některé odlišnosti, které se odvíjejí například od typů vazeb mezi uživateli těchto sítí. Například v rámci Facebooku mezi sebou uživatelé uzavírají oboustranné přátelství, takže graf obsahuje neorientované hrany, zatímco u Twitteru nebo Instagramu je možné sledovat jiné uživatele, aniž by oni sledování opětovali, tudíž jsou hrany grafu orientované.

1.2.2.2. Informační sítě

Druhou kategorií jsou sítě informační, které jsou také někdy nazývány jako znalostní. Klasickým příkladem patřícím do této kategorie jsou citační sítě. Ve vědeckých článcích se obvykle objevují citace na předchozí práce jiných autorů pojednávajících o stejném tématu. Tyto citace formují síť, ve které vrcholy reprezentují články a orientované hrany znázorňují, že článek, z něhož vychází hrana, cituje článek, do kterého hrana směřuje. Struktura takto vytvořené citační sítě znázorňuje strukturu informací uložených v jejích vrcholech. (2)

V článku mohou být pochopitelně citovány pouze starší články, nikoliv ty, které ještě nebyly napsány. Z tohoto důvodu všechny hrany citačních sítí směřují v čase do minulosti, a v těchto sítích se tak obvykle nenacházejí žádné kružnice. Graf je tedy acyklický. Výjimkou mohou být aktualizované online zdroje. (2)

Vytváření citačních sítí značně ulehčuje vysoký počet přesných zdrojů. Mezi dva často analyzované příklady informačních sítí patří citační síť akademických prací a World Wide Web (WWW), ve které hrany reprezentují odkazy mezi webovými stránkami. (2)

WWW svým rozsahem značně převyšuje všechny ostatní mapované sítě. Každý uzel této sítě reprezentuje webovou stránku, na kterou odkazuje několik jiných stránek a zároveň ona sama odkazuje na jiné webové stránky. (1)

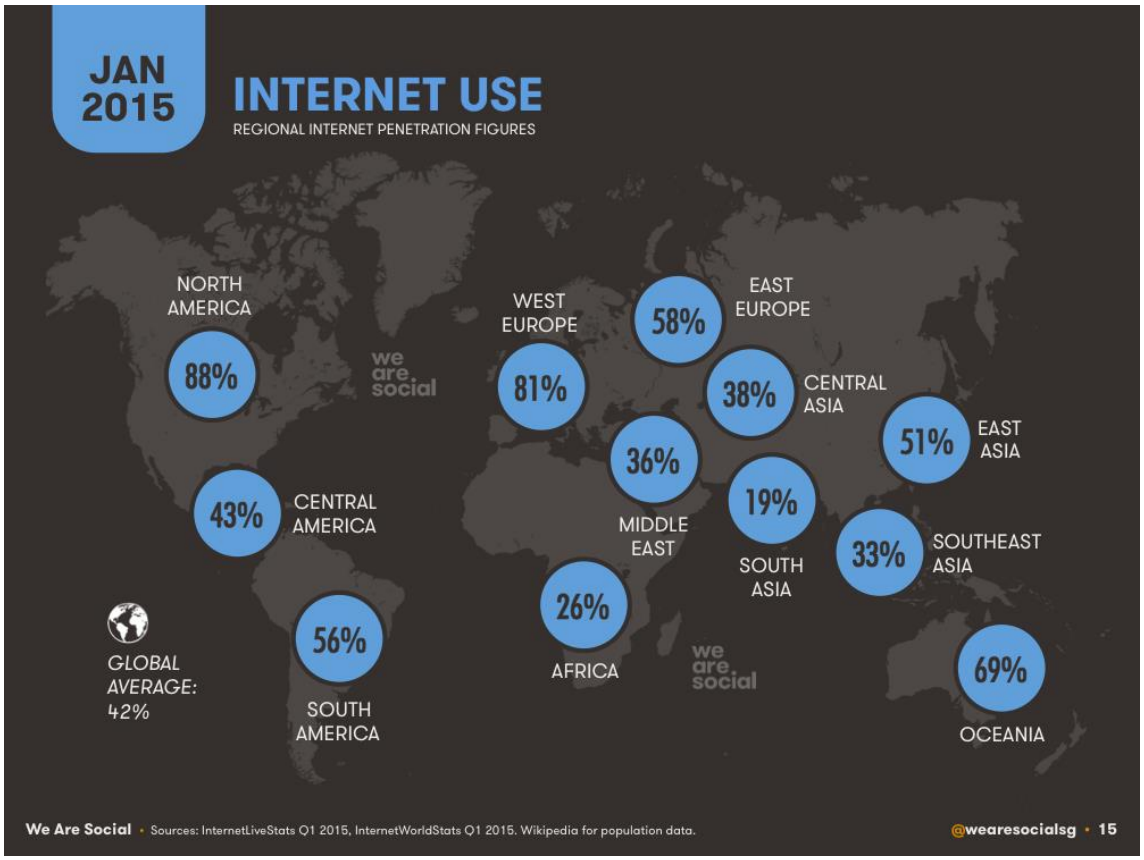
1.2.2.3. Technologické sítě

Dalším typem komplexních sítí jsou sítě technologické. Jedná se typicky o sítě navržené k distribuci nějakého zdroje, kterým může být například elektřina nebo informace. Do této kategorie patří také síť leteckých cest, síť silničních komunikací, železnic apod. (2)

Telefonická síť může být také z určitého pohledu považována za technologickou a to v případě, že není bráno v potaz, kdo komu volá, ale je zkoumána jako fyzická síť telefonů a spojujících kabelů. (2)

Také Internet je zařazen mezi technologické sítě. V této síti jsou znázorněna fyzická spojení mezi počítači. Vzhledem k rozsahu Internetu je struktura této sítě zkoumána z vyššího pohledu, ve kterém jsou počítače sdruženy do různě velkých skupin nebo také často pouze na určitém geografickém území. Internet je obvykle zkoumán ze dvou měřítek: na úrovni autonomních systémů nebo na úrovni routerů. Za autonomní systém jsou považovány skupiny řízené společnou správou. (1)

Internet a WWW jsou zajímavé sítě, jelikož díky své velikosti umožňují spolehlivou statistickou analýzu jejich topologických vlastností. Rozšířenost Internetu je dokumentována na obrázku číslo 3. Na druhou stranu, se prvky těchto sítí mohou ze stejného důvodu značně lišit. Například v rámci WWW je obsah každé stránky určen jejím majitelem. I přes tyto rozmanitosti lze i u těchto sítí pozorovat některé typické vlastnosti, jako je existence malého světa. (1)



Obrázek 3: Míra lidí využívajících Internet v jednotlivých částech světa, převzato z: http://wearesocial.sg/blog/2015/01/digital-social-mobile-2015/

1.3. Měření sítí

Zatímco malé sítě mohou být popsány grafem a mohou být znázorněny v jednoduchém obrázku, větší sítě může být složitější vizualizovat a popsat. Je tak nutné najít alternativním způsoby, kterými je možné sítě klasifikovat a vzájemně je porovnávat. Postupně tak vznikly různé statistiky a charakteristiky, které určují vlastnosti sítě. Díky těmto vlastnostem je možné udělat si určitou představu o struktuře i v případě rozsáhlých sítí. (3)

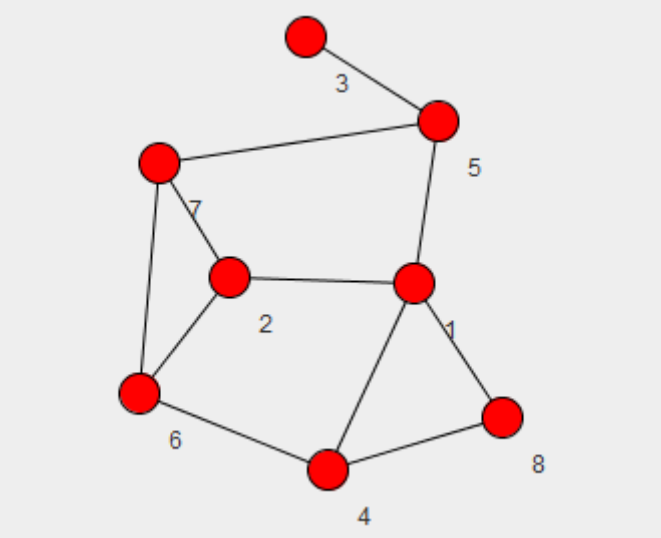
1.3.1. Hustota

Hustota sítě je definována jako průměrný počet sousedů všech uzlů v síti. Hodnota hustoty lze vypočítat tak, že se vydělí počet hran maximálním možným počtem hran. Jedná se tedy o procentuální vyjádření počtu hran z maximálního možného. Hustota indikuje míru propojenosti uzlů v rámci sítě. V případě prázdného grafu, kdy nejsou žádné uzly propojeny, nabývá hustota hodnoty 0. Naopak v kompletním grafu, kdy jsou vzájemně propojeny všechny vrcholy, nabývá hodnoty 1. (10)

Výrazný rozdíl v hustotě může být například mezi sítí přátelství lidí na malé vesnici v porovnání se sítí přátelství lidí žijících ve velkém městě. V případě vesnice má graf málo vrcholů a lze očekávat, že se vzájemně dobře zná většina obyvatel, takže graf má vysoký počet hran a hustota je také vysoká. Naopak v případě velkého města, ve kterém žije mnoho lidí, se dá předpokládat, že každý zná pouze malou část obyvatel města, a hustota sítě je tak nízká.

$$D = \frac{|e|}{\frac{n * (n-1)}{2}}$$

V rovnici pro výpočet hustoty je |e| počet hran grafu, „n“ znázorňuje počet vrcholů.



Obrázek 4: Ukázková síť pro výpočet statistik sítě

Síť na obrázku 4 obsahuje 8 vrcholů a 11 hran, takže výpočet hustoty vypadá takto:

$$D = \frac{11}{\frac{8 * (8-1)}{2}}.$$

Pomocí tohoto výpočtu lze zjistit, že hustota sítě je 0,39.

1.3.2. Průměrný stupeň

Stupeň vrcholu je počet hran, které do daného vrcholu zasahují. Statistika sítě průměrný stupeň označuje průměr stupňů všech vrcholů sítě. Tuto veličinu lze snadno vypočítat tak, že se vydělí dvojnásobek celkového počtu hran |e| celkovým počtem vrcholů „v“. (5)

$$AD = \frac{2 * |e|}{v}$$

Výpočet pro síť na obrázku 4 je následující: $AD = \frac{2*11}{8} = 2{,}75$. Tento výsledek je možné ověřit průzkumem jednotlivých vrcholů a jejich stupňů. Vrcholy 1 až 8 mají tyto stupně: 4, 3, 1, 3, 3, 3, 3, 2. Pokud se tyto hodnoty sečtou a vydělí počtem hran, který je 11, tak vyjde opět 2,75.

Hodnoty stupňů všech vrcholů sítě tvoří distribuční funkci, která poskytuje důležité informace o síti. Distribuce stupňů vrcholů sítě, která určuje pravděpodobnost, že náhodně vybraný uzel má k spojení, je podrobně zkoumána v různých druzích sítí. Pokud je distribuční funkce mocninná, což je velice časté v sítích reálného světa, tak je tato síť označována jako bezškálová. (11)

1.3.3. Průměrná délka cesty

Nejkratší cesta mezi dvěma uzly sítě je definována jako minimální počet hran, kterými je nutné projít při předání informace mezi těmito dvěma uzly. Další důležitou a často analyzovanou charakteristikou sítí je průměrná délka nejkratších cest. Ta je vypočítávána jako průměr délek nejkratších cest mezi všemi možnými variantami dvojic uzlů sítě. Tato míra určuje, jak efektivně jsou šířeny informace v síti. Vysoká časová náročnost algoritmu pro hledání nejkratších cest však neumožňuje výpočet této statistiky pro některé velice rozsáhlé sítě reálného světa. (12)

Většina sítí reálného světa má velmi krátké průměrné délky cest, takže informace v rámci sítě jsou sdíleny efektivně. V kontextu sítí malého světa to znamená, že každé dva uzly sítě jsou dobře propojeny a je mezi nimi krátká cesta. Pokud je z grafu odstraněna libovolná hrana, tak se snižuje propojenost sítě. (13)

Existuje několik algoritmů pro hledání nejkratších cest. Tyto algoritmy jsou v rámci této práce popsány v rámci kapitol věnujících se centralitě sítí.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	
1	-	1	2	1	1	2	2	1	
2	1	-	3	2	2	1	1	2	
3	2	3	-	3	1	3	2	3	
4	1	2	3	-	2	1	2	1	
5	1	2	1	2	-	2	1	2	
6	2	1	3	1	2	-	1	2	
7	2	1	2	2	1	1	-	3	
8	1	2	3	1	2	2	3	-	
AVG	1,43	1,71	2,43	1,71	1,57	1,71	1,71	2,00	**1,79**

Tabulka 1: Výpočet průměrné délky cesty sítě na obrázku 4

Tabulka 1 obsahuje hodnoty délky nejkratších cest mezi všemi dvojicemi uzlů sítě na obrázku 4. Na posledním řádku je vypočítaná průměrná délka cesty z příslušného uzlu do všech ostatních uzlů sítě. Průměr těchto hodnot určuje průměrnou délku cesty sítě. V tomto případě je průměrná délka cesty rovna 1,79. Jedná se pouze o ukázkovou síť malých rozměrů, takže délka průměrné cesty vychází velice nízká.

1.3.4. Průměr sítě

Průměr sítě (network diameter) sítě je definován jako délka nejdelší z nejkratších cest mezi všemi dvojicemi uzlů sítě. Algoritmus pro výpočet průměru sítě je tak stejně jako algoritmus pro výpočet délky průměrné cesty časově náročný, protože je nutné určit délku nejkratší cesty mezi všemi dvojicemi uzlů sítě. (14)

Mnoho sítí reálného světa má překvapivé malý průměr sítě. Tento fakt dokazují Milgramovy experimenty objevující fenomén malého světa a z nich vycházející koncept šesti kroků odloučenosti. (14)

Z tabulky 1, která obsahuje hodnoty délky nejkratších cest mezi všemi dvojicemi uzlů sítě, je možné vyčíst, že průměr sítě na obrázku 4 je roven 3, jelikož právě 3 je maximální hodnota tabulky.

1.3.5. Modularita

Modularita určuje kvalitu struktury seskupení neboli shlukování sítě. Shlukování sítě je považováno za kvalitní, pokud jsou uzly přiřazené do stejného shluku vzájemně pevně propojené a málo propojené s uzly patřícím do jiného shluku. Pomocí modularity sítě mohou být odhalovány struktury komunit v síti. (15)

Výpočet modularity je definován takto:

$$Q = \frac{1}{2m} * \sum_{i,j} \left[\left(A_{i,j} - \frac{k_i * k_j}{2m} \right) * \delta(c_i, c_j) \right].$$

V této formuli reprezentuje $\delta(c_i, c_j)$ Kroneckerovo delta, které je rovno 1, pokud vrcholy „i“ a „j“ patří do stejného shluku, v opačném případě je rovno 0. „A“ je matice sousednosti, jejíž prvky určují počet hran sítě, „k_i“ a „k_j“ jsou stupně vrcholů „i“ a „j“. Hodnota $\frac{k_i * k_j}{2m}$ vyjadřuje počet hran, který by byl očekávaný v případě aplikování náhodného rozložení hran v síti místo skutečného preferenčního rozložení, které způsobuje vznik struktury komunit. (15)

Hodnota modularity je v rozmezí 0 a 1. Čím větší je Q, tím lépe je síť rozdělena do komunit. Pokud se rozložení hran neliší od náhodného rozložení, je Q rovno 0. (16)

Pro výpočet modularity sítě na obrázku 4 je možné použít nástroj pro analýzu sítí Gephi. Pomocí tohoto nástroje byly v síti objeveny 3 shluky a podařilo se zjistit, že modularita sítě je rovna 0,269. Se znalostmi o modularitě sítě lze již podle obrázku odhadnout, že modularita této sítě bude malá, jelikož vrcholy sítě nejsou výrazně odděleny do více části a síť má strukturu blízkou náhodné, ve které jsou stupně vrcholů téměř rovnoměrné.

1.3.6. Koeficient shlukování

Sítě reálného světa vykazují výraznou tendenci ke shlukování uzlů, což znamená, že hustota blízkého okolí uzlu vyjádřena koeficientem shlukováním je velice často výrazně vyšší než celková hustota sítě. V sociálních sítích lze tento jev vysvětlit na příkladu sítě přátelství tak, že pokud má aktér dva přátele, tak je výrazně vyšší pravděpodobnost, že tyto dva přátelé spolu také tvoří přátelství, než pokud jsou dva lidé vybráni zcela náhodně. (17)

Koeficient shlukování (clustering coefficient) uzlu „v“ je definován jako pravděpodobnost, že dva náhodně vybraní sousedé uzlu „v“ jsou vzájemně propojeni. Výpočet koeficientu shlukování je technicky řešen jako počítání trojúhelníků v grafu. Pomocí hledání trojúhelníků lze zjistit, zda jsou dva sousední uzly také propojeny. V případě, že má uzel méně než dva sousední uzly, tak je jeho koeficient shlukování roven nule. Koeficient shlukování uzlu „v“ je definován jako počet spojení v jeho okolí, které obsahuje všechny sousední uzly, vydělený maximálním možným počtem spojení v tomto okolí. (17), (18)

$$CC(v) = \frac{|\{e_{jk}: v_j v_k \in N_i, e_{jk} \in E\}|}{\frac{k_v(k_v - 1)}{2}}$$

Jmenovatel zlomku vyjadřuje maximální možný počet hran v okolí, „k_v“ znázorňuje počet sousedních uzlů uzlu „v“. Čitatel zlomku vyjadřuje skutečný počet hran v okolí uzlu „v“. (17)

Koeficient shlukování daného uzlu je mikro míra, která charakterizuje pouze příslušný uzel. Pro celkovou charakteristiku sítě je používán globální průměrný koeficient

shlukování, který je vypočítáván jako průměr koeficientů shlukování všech uzlů sítě. (18)

Pro výpočet průměrného koeficientu shlukování sítě na obrázku 4 je nejprve nutné vypočítat koeficienty shlukování jednotlivých uzlů.

Uzel	1	2	3	4	5	6	7	8	
Počet sousedů	4	3	1	3	3	3	3	2	
Hrany v okolí	1	1	0	1	0	1	1	1	
Max. hran v okolí	6	3	0	3	3	3	3	1	Průměr
Koef. Shlukování	0,17	0,33	0,00	0,33	0,00	0,33	0,33	1,00	0,31

Tabulka 2: Výpočet průměrného koeficientu shlukování sítě na obrázku 4

Z tabulky 2 vyplývá, že nejvyšší koeficient shlukování 0,33 mají uzly „2“, „4“, „6“ a „7“. Naopak nejnižší koeficient shlukování 0 má uzel „5“ a uzel „3“, který je spojen pouze s jedním uzlem. Průměrný koeficient shlukování sítě je vypočítán jako průměr hodnot koeficientů shlukování všech uzlů a v tomto případě vychází 0,31.

1.3.7. Centralita

Mnohé studie ukázali, že v rámci komplexních sítí se obvykle výrazně liší stupně jednotlivých vrcholů. Jinými slovy lze tedy říct, že některé vrcholy sítě jsou často propojeny s velkým počtem jiných vrcholů, zatímco existují i vrcholy, které jsou v rámci sítě propojeny s daleko nižším počtem vrcholů. (9)

Určit, do které z těchto dvou skupin daný vrchol patří, pomáhá míra zvaná centralita. Jedná se na rozdíl od většiny ostatních měr o mikro míru. To znamená, že je určována z hlediska jednotlivých uzlů sítě a určuje jejich vztah k celé síti. Makro míry popisují sítě komplexně. (3)

Centralita porovnává uzly podle jejich vlivu v síti a určuje tak jejich důležitost. Jedná se o jeden z nejvíce zkoumaných konceptů v analýze sociálních sítí. Nejdůležitější aktéři sítě jsou obvykle strategicky umístěni v rámci sítě. Centralita může být kromě jednotlivých aktérů vypočítávána také pro určitou skupinu aktérů v síti. Pomocí skupinové míry je možné změřit, jak důležitá je seskupená sada aktérů jako celek. (19)

Neexistuje jednotný způsob, kterým se vyjadřuje míra centrality, různí autoři zvolili rozdílné přístupy k posouzení důležitosti aktéra v síti. Lze však říci, že všechny pojmy, které nějakým způsobem vyjadřují pozici uzlu v síti, jsou užitečné. Z tohoto důvodu vzniklo postupně několik různých měr centrality, které zachycují různé vlastnosti uzlu,

jež mohou být užitečné při sledování různého chování sítě. Míru centrality je vhodné zvolit podle povahy vztahů, které daná síť zkoumá. (3)

Míra centrality je pro každý uzel vyjádřena kladným číslem, které znázorňuje míru jeho propojenosti. Čím je číslo vyšší, tím je také propojenost vyšší. V případě dvou identických struktur sítě by měly mít uzly ne stejné pozici identickou hodnotu centrality. (9)

2. Přehled metrik centrality

Míry centrality mohou být rozděleny do čtyř hlavních skupin z hlediska typu statistik, na kterých jsou založeny. Tyto skupiny jsou následující:

- degree (stupeň),
- closeness (blízkost – jak snadno může uzel dosáhnout dalších uzlů),
- betweenness (jak důležitý je uzel z hlediska propojení ostatních uzlů),
- míry vypočítávané podle sousedních uzlů (jak důležité nebo vlivné jsou sousední uzly).

Již podle názvu jednotlivých skupin je možné odhadovat, že tyto metriky zachycují různé aspekty pozice uzlu. Je zřejmé, že pro různé příklady je vhodné využít různé metriky. Někdy může nabízet důležité údaje jedna metrika a v jiném případě může být vhodnější využít metriku jinou. Všechny tyto skupiny měr v základní podobě určují míru centrality u neorientovaných grafů bez ohodnocení. (3)

Různé metriky mají rozdílné přístupy k pojetí toku dat v rámci sítě. Některé míry, jako základní verze closeness a betweenness, počítají pouze geodetické cesty a předpokládají tak, že všechna data v síti se pohybuju ideálním způsobem, tedy po nejkratší možné cestě. Další míry, mezi které patří například random-walk closeness nebo κ-path varianta betweenness centrality nepředpokládají pouze nejkratší cesty, ale předpokládají cesty, ve kterých žádný vrchol není navštíven více než jednou. Poslední varianta, jak mohou metriky přistupovat k proudění dat, zahrnuju možnost, že data v rámci cesty mohou navštívit jeden vrchol i více než jednou. Do této skupiny metrik patří eigenvector nebo Katz centralita. Bez ohledu na trajektorii cesty, některé míry, jako betweenness, předpokládají, že data proudí od vrcholu do vrcholu po jedné vybrané cestě, zatímco například eigenvector umožňuje využití více cest zároveň. (20)

2.1. Degree centrality

Nejjednodušší mírou určující centralitu uzlu v sítí je degree centrality. Ta, jak název napovídá (degree = stupeň), je určena podle stupně uzlu. Její hodnota se tedy rovná počtu sousedních uzlů. Uzel stupně n-1, tedy propojený se všemi ostatními uzly sítě, má nejvyšší možnou centralitu v dané síti. (3)

Uzel s vysokým stupněm je v přímém kontaktu s mnoha dalšími uzly, takže by měl být ostatními uzly považovaný za důležitý prostředek pro komunikaci v síti a tedy za její

klíčový prvek, která se nachází v ústřední pozici. Opakem jsou uzly s nízkým stupněm, které jsou tak umístěny na periferiích sítě nejsou příliš aktivní v relačních procesech. V extrémním případě, kdy je uzel plně osamocený a jeho stupeň je roven nule, nemá jeho odstranění ze sítě žádný vliv na ostatní členy sítě. (19)

Degree centrality zanedbává několik zajímavých vlastností sítě z hlediska centrality. Zcela ignorovány jsou nepřímá spojení. Může se stát, že uzel má relativně nízký stupeň, ale leží v kritickém místě sítě. V tomto případě poskytují vhodnější výsledky jiné míry centrality. (9)

2.2. Míry vypočítávané podle nejkratších cest

Samostatnou skupin tvoří míry, které jsou vypočítávány pomocí nejkratších cest mezi vrcholy grafu.

2.2.1. Closeness centrality

Closeness centrality sleduje, jak blízko je daný uzel ke všem ostatním uzlům sítě. Je založena na myšlence, že uzly s kratší vzdáleností k ostatním uzlům mohou šířit informace v rámci sítě efektivněji. V tomto případě je tedy vrchol považovaný za důležitý a hodně propojený, pokud může dosáhnout všech ostatních vrcholů sítě přes nízký počet zprostředkovatelů a není příliš závislý na ostatních vrcholech. (9)

Closeness centrality je obvykle interpretována jako ukazatel předpokládaného času, který uběhne do doručení nějakých dat proudících v síti. Vrcholy s vysokou hodnotou closeness mají nižší vzdálenosti od ostatních vrcholů, a tak jsou jim data doručována dříve než vrcholům s nižší closeness centralitou. (20)

Ústřední uzly sítě potřebují malý počet kroků k dosažení ostatních uzlů, neboli cesta mezi nimi a ostatními uzly musí být co nejkratší. Closeness centralita je nepřímo úměrná vzdálenosti k ostatním uzlům sítě, občas je tak označována také jako „nejkratší vzdálenost“. Pokud se při změně v síti zvýší vzdálenost uzlu od uzlů ostatních, tak se sníží hodnota centrality, protože se zvýší počet hran na cestě spojující tento uzel s ostatními. (19)

Nedostatkem closeness centrality je, že je závislá pouze na nejkratších cestách, které nejsou vždy jedinou možností, jak se v rámci sítě mohou šířit informace. Tento problém se snaží řešit alternativní způsob výpočtu, který se nazývá random-walk closeness centrality. Tento výpočet používá místo nejkratších cest náhodné procházky, což jsou

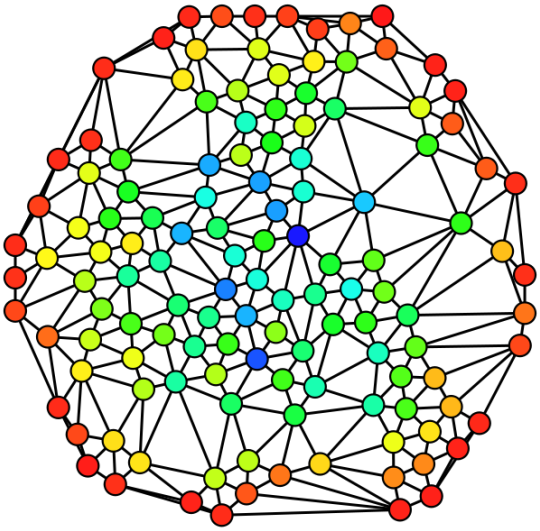
cesty, které vznikají tak, že v každém uzlu dochází k náhodnému výběru sousedního uzlu, kterému je informace dále předána. (21)

2.2.2. Betweenness centrality

Interakce mezi dvěma nesousedními aktéry sítě může záviset na dalších aktérech sítě. Zejména aktéři ležící na cestě mezi dvěma jinými aktéry mohou mít velký vliv. Tito aktéři mohou potenciálně mít určitou kontrolu nad interakcí dvou aktérů sítě, kteří spolu nejsou přímo propojeni. (19)

Betweenness centrality určuje hodnotu příslušící uzlu v závislosti na množství informací, které kontroluje. Předpokládá se, že komunikace a interakce mezi dvěma uzly, které nejsou přímo propojené, je ovlivněna zprostředkovateli. (9)

Tato centralita je založena na vyhodnocování toho, jak dobře je uzel umístěn v kontextu cest, na kterých leží. Hodnota této metriky vyjadřuje, jak je daný uzel důležitý pro spojení dvou jiných uzlů. Uzel je považovaný za dobře propojený, pokud se nachází na vysokém počtu nejkratších cest mezi dvěma uzly. (9)



Obrázek 5: Znázornění betweenness centrality

Na obrázku 5 je znázorněna betweenness centralita aktérů sítě. Vzhledem k tomu, že je použito správné a přehledné rozložení vrcholů, lze již podle umístění jednotlivých vrcholů odhadnout, které vrcholy leží na velkém počtu nejkratších cest mezi ostatními vrcholy a budou tak důležité podle betweenness. Vrcholy s nejnižší hodnotou betweenness mají červenou barvu, s rostoucí hodnotou se postupně barva mění podle barevného spektra až do barvy modré. Potvrzuje se tak, jak se dalo předpokládat, že za nejdůležitější jsou považovány vrcholy, které jsou na obrázku 5 vykresleny uprostřed

grafu. Naopak na periferiích grafu mají vrcholy červenou barvu, což znázorňuje nízkou hodnotu betweenness centrality.

2.3. Míry vypočítávané podle sousedních uzlů

Do poslední skupiny patří míry Eigenvector (vlastní vektor) a Katz. Tyto míry jsou složitější než předchozí, jejich myšlenka je založena na předpokladu, že důležitost uzlu je určena tím, jak důležité jsou jeho sousední uzly. Při výpočtu je tak zohledněn nejenom počet sousedních uzlů, ale především to, kolik ze sousedních uzlů je v rámci sítě důležitých. Na tomto smyslu jsou založeny citační žebříčky nebo Google page ranking (hodnocení webových stránek). (3)

Komplikací je, že tyto metriky jsou autoreferenční, tedy odkazují samy na sebe. Centralita uzlu je závislá na tom, jaká je centralita sousedních uzlů, jichž centralita je také závislá na sousedních uzlech, tedy i na uzlu původním. Existuje několik přístupů, jak se vyrovnat s tímto problémem. (3)

2.3.1. Eigenvector centrality

Eigenvector centrality je založena na myšlence, že spojení s více propojeným uzlem více přispívá pro zvýšení vlastní centrality, než spojení s méně propojeným uzlem. Jak bude v dalších kapitolách popsáno, tato míra se složitěji interpretuje a její výpočet je méně srozumitelný, než je tomu u dříve zmíněných měr. (9)

Eigenvector centralita má několik výhod oproti předchozím měrám. Zatímco degree, closeness i betweenness mohou mít pouze binární vztahy mezi vrcholy, eigenvector centralita může být úspěšně použita i v grafech s ohodnocením hran. V případě obyčejných grafů bez ohodnocení, kde může být využita i degree centrality, je značný rozdíl mezi návrhem těchto měr. Největší rozdíl v hodnotě centrality vzniká, pokud je vrchol propojen s mnoha vrcholy, které mají nízký stupeň, nebo pokud je naopak vrchol propojen s malým počtem vrcholů, které mají vysoký stupeň. (22)

Důležitým rozšířením je beta-centralita $c(\beta)$, která umožňuje přiřazení váhového koeficientu β. (22)

2.3.1.1. PageRank

PageRank je algoritmus vyvinutý v roce 1998, který používá Google k ohodnocení webových stránek. Má za úkol simulovat chování uživatelů surfujících na Internetu.

Tento algoritmus byl vytvořen v rámci vývoje webového vyhledávače. PageRank přiřazuje webovým stránkám důležitost podle počtu hypertextových odkazů, které na ni ukazují z jiných stránek. Jedná se o rozšířenou variantu eigenvector centrality. (23)

V rámci vývoje vyhledávače Google byla namodelována síť, která se skládá z webových stránek. Pokud stránka obsahuje hypertextový odkaz na stránku jinou, v grafu je to zaznamenáno orientovanou hranou. Jedná se tedy o orientovaný graf, který umožňuje výpočet hodnoty PageRank. Tato hodnota vyjadřuje objektivní citační důležitosti, která se pokouší co nejvíce korespondovat s preferencemi uživatelů. Pomocí této hodnoty se vyhodnocují řetězce zadané uživateli do vyhledávače, takže nejpopulárnější weby mají nejvyšší prioritu a jsou ve výsledku uvedeny na prvních místech. (24)

2.3.2. Katz centrality

Jeden z prvních návrhů, jak posuzovat centralitu autoreferenční metodou, byl představen Leo Katzem. Tento návrh je založen na myšlence, že k určení důležitosti individuality v sociální síti není dostačující počítat pouze přímá spojení. Pokud má aktér „i“ spojení pouze s aktéry „k“ a „l“, ale všichni ostatní aktéři v síti jsou spojeni s jedním ze dvojice aktérů „k“, „l“, tak může být aktér „i“ nejdůležitější v síti, i přesto že má pouze dvě přímá spojení. Se všemi ostatními aktéry je spojen nepřímo. (25)

Podle Katze hraje tedy důležitou roli z hlediska propojenosti uzlu v sítích nejen počet přímých připojení, ale také propojenost sousedních uzlů. Katz zahrnuje do kalkulace všechny cesty libovolné délky ze zkoumaného uzlu do všech ostatních uzlů sítě. (9)

3. Způsoby výpočtu centrality

V této kapitole jsou uvedeny a prozkoumány způsoby výpočtu zmíněných měr centrality.

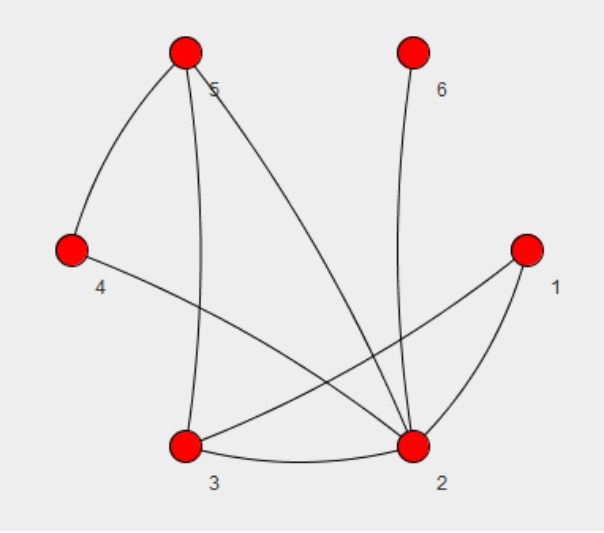
3.1. Degree centrality

Jak už bylo dříve zmíněno, vypočet degree centrality je velice jednoduchý. V tomto případě je propojenost příslušného uzlu určena pouze v závislosti na počtu jeho přímých sousedů. Hodnota centrality uzlu je tedy tím vyšší, čím je vyšší počet přímo připojených jiných uzlů. (9)

$$DC(x) = \deg(x)$$

Vzhledem k tomu, že tato míra obsahuje pouze výpočet přímých sousedů n uzlů sítě, má její výpočet složitost O(n). (9)

V tabulce 3 jsou uvedeny hodnoty degree centrality sítě z obrázku 6. V tabulce je také uvedeno pořadí propojenosti uzlů. Uzlem s nejvyšší hodnotou degree centrality je uzel 2, který má stupeň a zároveň tedy i degree centrality 5.



Obrázek 6: Příklad sítě, na kterém je ukázán výpočet degree centrality

Uzel	1	2	3	4	5	6
DC	2	5	3	2	3	1
Pořadí	4.	1.	2.	4.	2.	6.

Tabulka 3: Výpočet degree centrality

3.1.1. Optimalizace výpočtu degree centrality

Vzhledem k jednoduchosti výpočtu degree centrality neexistuje a ani není třeba mnoho optimalizačních metod. K dosažení rychlejšího výpočtu je vhodné vytvořit pole, které obsahuje hodnoty stupňů všech vrcholů grafu. Pro zjištění degree centrality vrcholu grafu tak stačí nalézt příslušnou hodnotu v tomto poli a doba výpočtu je tak pro všechny vrcholy konstantní. (26)

3.2. Closeness centrality

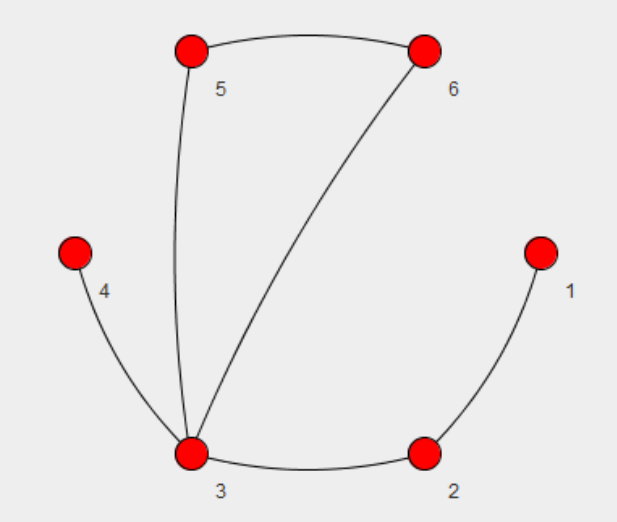
Při výpočtu closeness centrality daného uzlu jsou sečteny vzdálenosti mezi tímto uzlem a všemi dalšími uzly sítě. Vzdálenost z vrcholu do vrcholu je definována jako nejnižší možný počet hran, kterými je nutné projít při výměně informací mezi těmito dvěma vrcholy. Je nutné použít převrácenou hodnotu, tak aby s rostoucí sumou vzdáleností klesala centralita. Jestliže při přidání propojení dvou uzlů dojde ke snížení vzdálenosti k některému uzlu, zvýší se hodnota centrality. (9)

$$CC(x) = \frac{1}{\sum_{i=1}^{n} d(x,i)}$$

Maximum closeness centrality, kterého dosáhne uzel přímo propojený se všemi ostatními uzly sítě, je rovno $(n-1)^{-1}$. Minimální dosažitelná hodnota je číslo blížící se k nule. Pokud je closeness rovna nule, tak daný uzel není dosažitelný všemi ostatními uzly. Uzel je považovaný za dosažitelný jiným uzlem, pokud existuje cesta, která je propojuje, v opačném případě uzly nejsou vzájemně dosažitelné. Closeness centralita je tak smysluplná a využitelná pouze za předpokladu, že graf sítě je souvislý. V opačném případě, kdy by graf měl více než jednu komponentu, by neexistovala cesta mezi všemi uzly sítě. (19)

V tabulce 5 jsou výsledky closeness centrality pro všechny uzly ze sítě na obrázku 7. Při výpočtu je nutné nejprve vypočítat sumy vzdáleností k ostatním uzlům, ty jsou uvedeny v tabulce 4. V případě takto malé sítě je možné určit vzdálenosti z obrázku, algoritmus výpočtu pro rozsáhlejší sítě je uveden v popisu aplikace.

Nejvyšší hodnotu centrality 1/6 má uzel 3. Jeho vzdálenosti k dalším uzlům jsou 2, 1, 1, 1, 1. Součet těchto hodnot je roven 6, takže jeho closeness centrality je 1/6.



Obrázek 7: Příklad sítě, na kterém je ukázán výpočet closeness centrality

	1	2	3	4	5	6	Suma
1		1	2	3	3	3	**12**
2	1		1	2	2	2	**8**
3	2	1		1	1	1	**6**
4	3	2	1		2	2	**10**
5	3	2	1	2		1	**9**
6	3	2	1	2	1		**9**

Tabulka 4: Výpočty vzdáleností mezi uzly

Uzel	1	2	3	4	5	6
CC	1/12	1/8	1/6	1/10	1/9	1/9
Pořadí	6.	2.	1.	5.	3.	3.

Tabulka 5: Výpočet closeness centrality

Existuje několik modifikací výpočtu closeness centrality. Jedním z nich je míra, která je vyjádřena jako inverzní hodnota průměru vzdáleností mezi uzlem a všemi ostatními uzly. Dalším způsob, jak vyjadřovat centralitu na základě blízkosti uzlů, je využít decay parametr $\delta,\ kde\ 0 > \delta > 1$. Tento parametr vyjadřuje váhu. (3)

3.2.1. Optimalizace výpočtu closeness centrality

Jednou z největších výzev současnosti v oblasti analýzy sociálních sítí je zpracování dynamických dat. Sítě reálného světa se stále vyvíjejí, což způsobuje nutnost stále aktualizovat příslušný graf reprezentující tuto síť. Neustále se také mění výpočty důležitých metrik této sítě, a tak je důležité, aby tyto výpočty probíhaly rychle.

V případě closeness a betweenness centrality způsobuje největší problém při výpočtu centrality hledání nejkratších všech cest mezi všemi vrcholy sítě, které je časově

náročné. Je tak důležité věnovat se optimalizaci těchto operací a pokusit se minimalizovat čas nutný k jejich provedení. (27)

V rámci výpočtu closeness centrality je nutné určit délku nejkratších cest mezi všemi vrcholy sítě. Existuje několik algoritmů, které řeší tuto problematiku, jejich časová složitost je $O(nm + n^2 \log n)$, kde n je počet vrcholů a m je počet hran grafu. Tyto algoritmy tak nejsou efektivní a u rozsáhlých sítí může čas nutný pro výpočet closeness centrality dosáhnout neúnosných mezí, existují tak některé optimalizační algoritmy, které urychlují výpočet mnohdy i za cenu snížení přesnosti výpočtů nebo jiných nevýhod. Mezi ty patří například Eppsteinův a Wangův aproximační algoritmus, který umí vypočítat closeness centrality se složitostí $O(\frac{\log n}{\epsilon^2}(n \log n + m))$, s aditivní chybou $\epsilon\Delta$ pro inverzní hodnotu centrality s pravděpodobností nejméně $1 - \frac{1}{n}$, kde $\epsilon > 0$ a Δ je průměrná hodnota centralit grafu. (28)

Někdy může být cílem určit hodnotu centrality pouze pro významné vrcholy sítě, což může výrazně urychlit výpočet. Některé z optimalizačních výpočtů closeness centrality tak určují významné vrcholy sítě a vypočítají pouze jejich hodnotu centrality. (28)

3.3. Betweenness centrality

V případě betweenness centrality je určována pravděpodobnost, že v případě komunikace mezi uzly „i“ a „j“ je použita cesta procházející uzlem „x“. Předpokládá se, že hrany mají stejnou váhu a informace je poslána nejkratší možnou cestou. Za nejkratší možnou cestu je považována tak, která obsahuje nejmenší možný počet vrcholů. Pokud je mezi uzlem „i“ a „j“ více cest se stejnou délkou, tak má každá z těchto cest stejnou pravděpodobnost, že bude zvolena. (19)

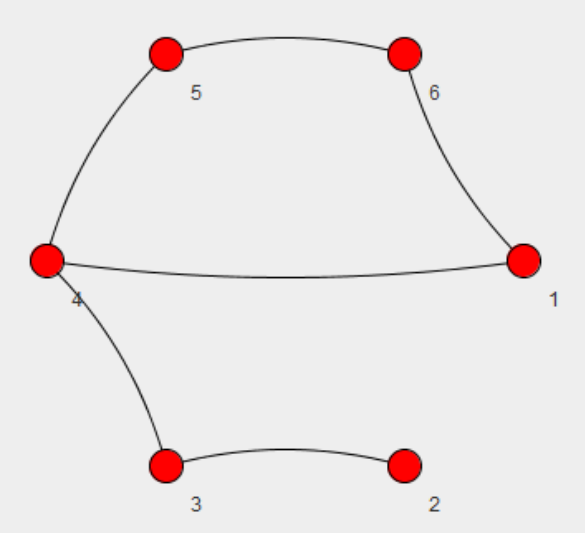
Základem výpočtu betweenness centrality je poměr počtu nejkratších cest mezi dvěma uzly procházejících přes sledovaný uzel a počtu všech nejkratších cest mezi těmito dvěma uzly. Tento výpočet ukazuje, nakolik je uzel důležitý z hlediska propojení dvou jiných uzlů. Čím více se zmíněný poměr blíží k hodnotě 1, tím je sledovaný uzel důležitější. (3)

Při výpočtu betweenness centrality uzlu je tedy nutné postupně vypočítávat tento poměr mezi všemi ostatními uzly sítě. V následující rovnici výpočtu centrality uzlu x vyjadřuje $g_{ij}(x)$ počet nejkratších cest mezi uzly i a j procházejících přes uzel x a g_{ij} vyjadřuje počet všech nejkratších cest mezi i a j. (9)

$$BC(x) = \sum_{i=1, i \neq x}^{n} \sum_{j=1, j<i, j \neq x}^{n} \frac{g_{ij}(x)}{g_{ij}}$$

Složitost výpočtu betweenness centrality podle Brandesova algortimu, který je využitelný pouze pro graf bez ohodnocení hran, je $O(n * m)$, kde n je počet uzlů sítě a m je počet hran sítě. (9)

V tabulce 7 jsou výsledky betweenness centrality pro všechny uzly ze sítě na obrázku 8. Tabulka 6 obsahuje ukázkový výpočet pro, z hlediska betweenness centrality, nejvíce propojený uzel 4. V této tabulce jsou uvedeny poměry nejkratších cest mezi všemi ostatními uzly sítě vedoucích přes uzel 4 a celkového počtu všech nejkratších cest mezi těmito dvěma uzly. Například mezi uzly 1 a 5 se nacházejí 2 nejkratší cesty, z nichž jedna prochází uzlem 4, takže poměr je 0,5. Hodnota centrality je vypočítána jako suma všech poměrů. V tabulce jsou poměry mezi uzly uvedeny vždy dvakrát, takže je nutné sumu vydělit dvěma.



Obrázek 8: Příklad sítě, na kterém je ukázán výpočet betweenness centrality

Uzel 4	1	2	3	5	6
1	-	1	1	0,5	0
2	1	-	0	1	1
3	1	0	-	1	1
5	0,5	1	1	-	0
6	0	1	1	0	-
				Suma	6,5

Tabulka 6: Výpočet poměrů nejkratších cest procházejících uzlem 4

Uzel	1	2	3	4	5	6
BC	1,5	0,0	4,0	6,5	1,5	0,5
Pořadí	3.	6.	2.	1.	3.	5.

Tabulka 7: Výpočet betweenness centrality

3.3.1. Optimalizace výpočtu betweenness centrality

Stejně jako v případě closeness centrality je vhodné zabývat se především optimalizací algoritmu pro hledání nejkratších cest, který je v případě betweenness centrality ještě komplikovanější, jelikož nestačí zjistit délku nejkratší cesty mezi uzly, ale musí se najít všechny nejkratší cesty mezi nimi.

Jednou z možností, jak dosáhnout toho, aby bylo možné počítat betweenness centrality i v rozsáhlých sítích je využít efektivnější algoritmus nazvaný kappa-path, který každému vrcholu grafu přiřazuje hodnotu κ-path. Vrcholy s vysokou hodnotou κ-path mají také vysokou betweenness centrality. Tato metoda má časovou složitost $O(k^3 n^{2-2\alpha} \log n)$ a každému vrcholu přiřazuje odhad κ-path hodnoty a její chybu $\pm n^{\frac{1}{2}+\alpha}$ s pravděpodobností $1 - \frac{1}{n^2}$. (29)

Centralita κ-path je založena na podobném předpokladu jako další varianta betweenness centrality random-walk. Myšlenka random-walk spočívá v procházení sítě s využitím tzv. „náhodných procházek“. Zpráva, která byla zaslána zdrojovým vrcholem, cestuje sítí po určité cestě s úmyslem dosáhnout cílového vrcholu. Všechny vrcholy sítě mají pouze jejich lokální pohled na síť, znají tedy pouze jejich sousední vrcholy. Pokud vrchol obdrží zprávu, tak se musí rozhodnout pouze podle jeho lokálního pohledu na síť, kterému vrcholu zprávu předá. Zprávu tedy pošle jednomu náhodně vybranému ze sousedních vrcholů. Zpráva takto stále pokračuje v cestování sítí, dokud nedorazí do cílového vrcholu. (29)

V případě κ-path je nutné přidat další dva předpoklady, pomocí kterých je snížen výpočetní čas bez výrazného odchýlení od principu random-walk. Zaprvé, zpráva při průchodu sítí nenavštíví žádný vrchol dvakrát, průchod je tedy ukončen, pokud je zpráva ve vrcholu, jehož všechny sousední vrcholy již byly v rámci putování zprávy navštíveny. Zpráva tedy musí nést informace o již navštívených vrcholech. Zadruhé, jelikož zpráva v sociálních sítích obvykle projde jen malým počtem hran, tak je zadaný parametr κ, který určuje právě maximální počet hran cesty, který nesmí být překročen. (29)

Na základě těchto předpokladů zní definice κ-path centrality takto: pro každý vrchol „v“ grafu $G = (V, E)$, je definována κ-path centralita $C_\kappa(v)$ vrcholu „v“ jako suma všech možných zdrojových vrcholů „s“ s pravděpodobností, že zpráva pocházející z „s“ projde „v“. Experimenty na reálných sítích ukázaly, že tato metoda zvyšuje přesnost detekování vrcholů s vysokou betweenness centrality a výrazně snižuje čas výpočtu v porovnání s ostatními algoritmy. (29)

Jiné optimalizační metody se snaží zvýšit rychlost výpočtu pomocí zlepšených struktur dat. Zatímco výpočetní složitost má algoritmus pro určení betweenness centrality vysokou, požadavky na paměť velké nejsou. To motivuje k využití paralelismu. (30)

3.4. Eigenvector centrality

Bonaci přišel s nápadem využít vlastní vektor největšího vlastního čísla matice sousedností pro posouzení centrality vrcholu v rámci sítě. Eigenvector centralita může být také brána jako vážená suma nejen přímých spojení, ale také nepřímých spojení všech délek, takže je do výpočtu zahrnuta celá síť.

Hodnota eigenvector centrality je získávána z vlastního vektoru v matice sousednosti grafu.

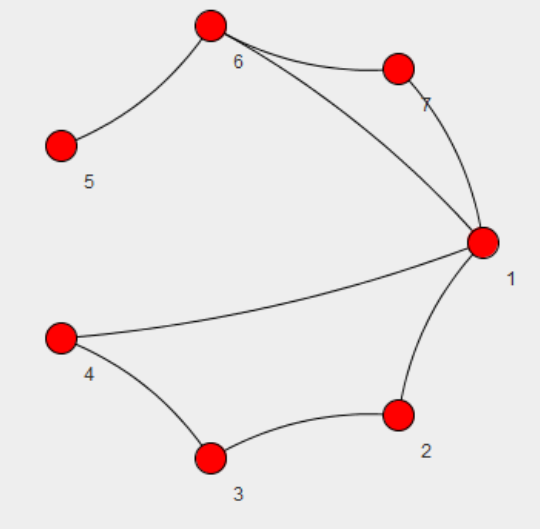
$$EC(x) = v_x = \frac{1}{\lambda_{max}(A)} \sum_{j=1}^{n} a_{jx} v_j$$

A je matice sousednosti, ve které je a_{ij} rovno 1, pokud je vrchol a propojený s vrcholem j, v opačném případě je tato hodnota rovna 0, v je vlastní vektor matice a a v_x je prvek tohoto vektoru, který náleží vrcholu x. Správné řešení musí mít pouze nezáporné hodnoty. Podle Perron-Frobeniovy věty je takové řešení právě jedno a to takové, ke kterému náleží největší vlastní číslo λ. (9)

Složitost výpočtu Eigenvector centrality je $O(n^2)$. (9)

3.4.1. Výpočet pomocí Excel doplňku matrix.xla

Excel neobsahuje funkce pro výpočet vlastních čísel a vektorů, pro jejich výpočet je nutné doinstalovat Matrix and Linear Algebra Package For Excel (matrix.xla). Jedná se o doplněk Excelu vytvořený na Michigan State University, který obsahuje funkce pro práci s maticemi.



Obrázek 9: Příklad sítě, na kterém je ukázán výpočet Eigenvector centrality a Katz centrality

Pro výpočet Eigenvector centrality je nejprve nutné vytvořit matici sousednosti. Ta vypadá pro síť na obrázku 9 následovně:

A	1	2	3	4	5	6	7
1	0	1	0	1	0	1	1
2	1	0	1	0	0	0	0
3	0	1	0	1	0	0	0
4	1	0	1	0	0	0	0
5	0	0	0	0	0	1	0
6	1	0	0	0	1	0	1
7	1	0	0	0	0	1	0

Tabulka 8: Matice sousednosti

Z této matice již lze pomocí funkce MatEigenvalue_max určit maximální vlastní číslo matice, které v tomto případě vyjde 2,57. Příslušný vektor vypočítá funkce MatEigenvector_inv, která má argumenty matici sousednosti a maximální vlastní číslo. Tento vektor obsahuje hodnoty, které jsou zároveň hodnotami Eigenvector centrality příslušných uzlů sítě. (31)

Uzel	1	2	3	4	5	6	7
EC	0,58	0,32	0,25	0,32	0,17	0,45	0,40
Pořadí	1.	4.	6.	4.	7.	2.	3.

Tabulka 9: Výpočet Eigenvector centrality

3.4.2. Vlastní čísla a vlastní vektory

Vlastní čísla (eigenvalues) a vlastní vektory (eigenvectors) se vyskytují ve studiích diferenciálních rovnic, které jsou používány při modelování vibrací ve vědě

a strojírenství. Mezi příklady patřící do této skupiny patří všechny možné vibrace, například strun, mostů, letadel atd. (32)

Je dána čtvercová matice A, ke které se hledají vlastní hodnoty λ, ke kterým přísluší vlastní vektor x. Platí tento vztah: $Ax = \lambda x$. (32)

U příkladu výpočtu Eigenvector centrality tedy platí, že součin matice sousednosti a vlastního vektoru je rovný součinu vlastního čísla a vlastního vektoru.

Ax	λx
1,49	1,49
0,83	0,83
0,65	0,65
0,83	0,83
0,45	0,45
1,15	1,15
1,03	1,03

Tabulka 10: Ověření výpočtu Eigenvector centrality

3.4.3. Optimalizace výpočtu eigenvector centrality

Lze říci, že eigenvector centrality je vlastně rekurzivní verzí degree centrality. Její hodnoty lze získat pomocí tohoto iteračního algoritmu:

- všem vrcholům je přiřazena hodnota centrality 1,
- centralita je přepočítána jako suma centralit všech sousedních vrcholů,
- hodnoty centrality jsou normalizovány tak, že jsou všechny vyděleny nejvyšší hodnotou z nich,
- předchozí dva kroky jsou opakovány, dokud nejsou hodnoty ustáleny.

Vrchol je považovaný za významný, pokud je propojený s jinými významnými uzly. Vrchol, který má vysokou hodnotu eigenvector centrality, je propojen s mnoha vrcholy, které mají také vysoký počet spojení s dalšími vrcholy. (33)

3.5. Katz centrality

Výpočet Katz centrality je definován následujícím vztahem, ve kterém 1 značí vektor n × 1, který obsahuje pouze čísla 1, e_x jednotkový vektor, k libovolný váhový faktor. Prvek a_{xy} matice A^i reprezentuje počet cest délky i z vrcholu x do vrcholu y. (9)

$$KC(x) = 1^T(\sum_{i=1}^{\infty} k^i A^i)e_x$$

K zajištění konvergence výpočtu musí být k menší než převrácená hodnota největšího vlastního čísla matice sousednosti $\lambda_{max}(A)$. Díky tomuto tvrzení je možné zjednodušit výpočet následujícím způsobem. (9)

$$KC(x) = 1^T((I_n - kA)^{-1} - I_n)e_x$$

I_n označuje Jednotkovou matici rozměru n=$|V_G|$. (9)

Váhový faktor „k“ může být také interpretován jako pravděpodobnost, že jeden vztah je užitečný pro vrchol x. (9)

Pro výpočet vektoru obsahujícího hodnoty Katz centrality pro všechny uzly sítě na obrázku 9 je tedy nutné v Excelu zadat následující složenou funkci: $\{= SOUČIN.MATIC(ones; INVERZE(I - k * A) - I)\}$, kde I je jednotková matice odpovídajících rozměrů, a je matice sousednosti, „k“ je váhový faktor a ones je řádkový vektor skládající se z hodnot 1. Výsledek je uveden v tabulce 11.

Uzel	1	2	3	4	5	6	7
KC	4,96	2,77	2,26	2,77	1,44	3,79	3,23
Pořadí	1.	4.	6.	4.	7.	2.	3.

Tabulka 11: Výpočet Katz Centrality

3.5.1. Optimalizace výpočtu Katz centrality

Vztah pro výpočet Katz centrality zahrnuje výpočet inverzní matice. Ta se obvykle získává pomocí LU rozkladu a následného využití Gausovy eliminační metody. Celková složitost těchto operací je $O(n^3)$, což způsobuje, že výpočet Katz centrality je mnohdy nepříjemně zdlouhavý. Nicméně tato složitost může být zredukována pomocí algoritmu Copperstmitha a Winogradova, který urychluje násobení matic, na $O(n^{2,376})$. (9)

Pro výpočet je také možné využít následující metodu, která odstraňuje výpočet inverzní matice, což značně sníží počet časově náročných maticových operací. Pomocí této metody je složitost výpočtu snížena na $O(n + m)$, kde m značí počet hran v síti. (34)

Je možné využít této geometrické řady:

$$(I - kA)^{-1} = I + kA + (kA)^2 + (kA)^3 + \cdots$$

Za předpokladu, že tato řada konverguje, je možné vypočítat vektor „s“ obsahující hodnoty Katz centrality podle jednoduché iterační metody produkující sekvenci odhadů s_0, s_1, s_2 ... následujícím způsobem:

$$s_0 = kAu$$

$$s_{k+1} = kAs_k + s_0, \text{ kde } k \geq 0$$

Výpočet může být ukončen, pokud se rozdíl výsledku dvou po sobě následujících iterací blíží nule. (34)

4. Popis aplikace

V rámci této práce byla napsána aplikace **Míry centrality**, ve které jsou implementovány algoritmy pro výpočet Degree centrality, Closeness centrality, Betweenness centrality, Eigenvector centrality i Katz centrality. Úkolem této aplikace je umožnit uživateli zadat libovolnou síť, pro kterou aplikace následně vypočítá hodnoty požadované míry centrality. Aplikace je napsána v jazyce Java.

4.1. Využité knihovny

Při vývoji aplikace bylo nutné vyřešit i některé problémy, které nejsou klíčové z hlediska této práce. K jejich rychlejšímu vyřešení byly využity knihovny Jung a Colt. Algoritmy pro výpočet jednotlivých měr centrality, jejichž součástí jsou mimo jiné také algoritmy pro hledání nejkratších cest, byly implementovány bez pomoci knihoven třetích stran, tak aby mohly být důkladně vysvětleny v této práci.

4.1.1. JUNG

JUNG je zkratka názvu Java Universal Network/Graph Framework. Tato knihovna (dostupná na *http://jung.sourceforge.net*) umožňuje modelování, analýzu a vizualizaci dat, které mohou být reprezentovány grafem nebo sítí.

První verze JUNG 1.0.0 byla vydána v srpnu 2003. Verze nejaktuálnější, která je použita i v této aplikaci, byla vydána v lednu roku 2010.

Součástí JUNG jsou některé algoritmy z teorie grafů, data miningu a analýzy sociálních sítí. Částečně se zabývá také vyhodnocováním centrality, implementovány jsou však pouze částečně jen některé míry.

V aplikaci Míry centrality je knihovna JUNG použita především pro vizuální znázornění grafu. Všechny grafy sítí v této práci pocházejí z vytvořené aplikace a jsou tedy vytvořeny pomocí knihovny JUNG, která zároveň umožňuje uživatelům graf upravovat.

4.1.2. Colt

Knihovna Colt (dostupná na *http://dst.lbl.gov/ACSSoftware/colt*) slouží k vědeckým a technickým výpočtům. Nové verze byly vydávány od roku 2000 do roku 2004.

Colt efektivně řeší problémy týkající se Matematiky, Statistiky, Lineární algebry a další.

Ve vytvořené aplikaci je tato knihovna používána k násobení matic a také k výpočtu inverzní matice, který je použit v algoritmu pro výpočet Katz centrality. Colt je použita také při výpočtu vlastních čísel a vlastních vektorů.

4.2. Layouty grafu

Vykreslení grafu pomáhá vytvořit lepší představu struktury vztahů mezi sledovanými objekty. Vzhledem k tomu, že datové struktury zachycované grafem mají mnohdy obrovské rozměry, libovolné vykreslení grafu nestačí, je nutné klást důraz na to, jakým způsobem k vizualizaci grafu dochází. Při konstrukci grafu s velkým počtem vrcholů a jejich spojení je nutné zvolit správný layout, tak aby se v něm bylo možné orientovat a byl přehledný. Je nutné najít správné techniky a algoritmy, podle kterých jsou vrcholy grafu rozmístěny. (35)

Uzly umístěné blízko sebe jsou uživateli chápány tak, že mají bližší vztah, i když nejsou propojeny. To znamená, že layout a uspořádání uzlů silně ovlivňuje, jak uživatel vnímá vztahy v grafu. Pro prezentaci grafu uživateli je tedy výběr layoutu klíčový. Volba závisí na tom, co má být v grafu zvýrazněno. (36)

Existují dva hlavní přístupy k vykreslování grafu. První klade velký důraz na algoritmy založené na matematické teorii grafů, druhý přístup vizualizace grafů je více interaktivní a více se soustředí na aplikační stránku. (36)

Existuje několik zásad a pravidel, kterými je vhodné se řídit při snaze sestrojit přehledný graf. Pokud se jedná o obyčejný dvourozměrný neorientovaný graf, je vhodné se při konstrukci grafu snažit dosáhnout zejména těchto bodů:

- graf by měl být souměrný,
- hrany by se neměly křížit,
- hrany by neměly mít ohyby,
- všechny hrany by měly být stejně dlouhé,
- rozdělení vrcholů by mělo být rovnoměrné. (37)

4.2.1. Layouty knihovny Jung

V aplikaci Míry centrality je graf vykreslován pomocí knihovny Jung, ve které je implementováno několik layoutů pro vizualizaci grafu.

4.2.1.1. Circle layout

V rámci tohoto layoutu jsou vrcholy uspořádány rovnoměrně do kruhu.

4.2.1.2. FR layout

Implementuje Fruchterman-Reingold algoritmus, který je definovaný následujícími třemi parametry.

- Násobitel přitažlivost: určuje, jak blízko se snaží hrany mít své vrcholy u sebe,
- Násobitel odporu: určuje, do jaké míry se snaží vrcholy vzájemné oddalovat,
- maximální počet iterací: číslo určující, kolikrát může algoritmus proběhnout před zastavením. (38)

Tento layout se tedy při vykreslování grafu řídí dvěma principy. Vrcholy propojené hranou by měly být vykresleny blízko sebe a žádné vrcholy by neměly být vykresleny příliš blízko u sebe. Vzdálenost vrcholů závisí na jejich počtu a na tom, kolik je pro vykreslení grafu dostupného prostoru. (38)
Jako hlavní výhoda tohoto algoritmu je uváděna rychlost. Tato výhoda se projevuje zejména u rozsáhlých grafů. FR layout se snaží o to, aby byl flexibilní a bylo ho možné použít pro téměř každý graf s uspokojivým výsledkem, aniž by byl uživatel nucen měnit další nastavení. (38)

4.2.1.3. ISOM layout

Metoda pro tvorbu ISOM layoutu je rozšířením Kohonenovy samoorganizující mapy, která je vhodná pro využití ve shlukové analýze. Jedná se o biologicky inspirovaný algoritmus, založený na umělých neuronových sítích. Jde o jednovrstvou umělou neuronovou síť, která umožňuje vizualizovat topografii a hierarchickou strukturu multidimenzionálních dat transformací do prostoru nižší dimenze. (39)

Hlavní výhodou tohoto přístupu je flexibilita a schopnost adaptace pro libovolné typy vizualizačního prostoru. Tato metoda spotřebovává relativně málo výpočetních prostředků a nevyžaduje žádné výrazné předzpracování. ISOM layout je možné použít i pro 3D zobrazení. (40)

4.2.1.4. KK layout

Tento layout Implementuje algoritmus Kamady a Kawaie (Kamada-Kawai algorithm for node layout). Jedná se o poměrně jednoduchou, ale přesto úspěšnou, metodu pro vykreslování neorientovaných grafů. (41)

Vzdálenost vrcholů je určena podle jejich euklidovské vzdálenosti. Graf je vykreslen pomocí systému, ve kterém jsou všechny dvojice vrcholů propojeny pomyslnou pružinou příslušné délky. Za optimální rozložení vrcholů je poté považován stav, ve kterém je součet energií ve všech pružinách nejnižší. (41)

Mezi dobré vlastnosti tohoto algoritmu patří symetrické vykreslení symetrických grafů, téměř shodné vykreslen isomorfních grafů, jednotná distribuce vrcholů nebo relativně malý počet překřížení hran. (41)

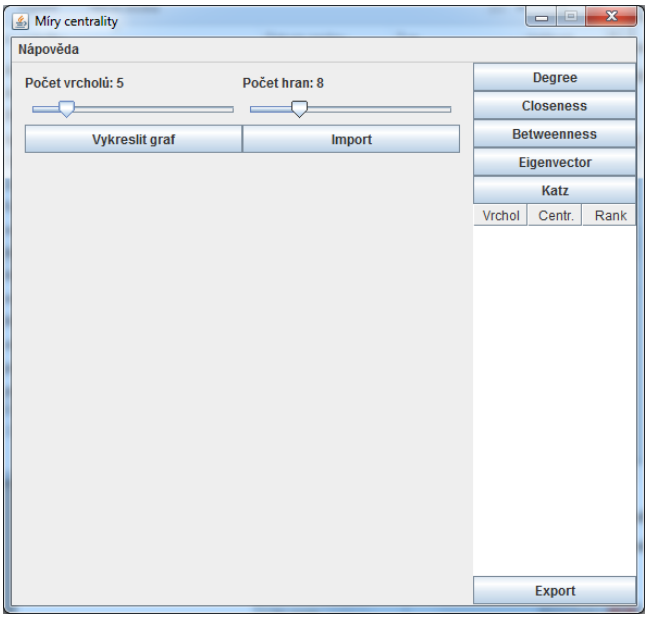
4.3. Náhodné generování grafu

Existuje několik známých standardně používaných generátorů náhodných grafů. Mezi ty patří například BarabasiAlbertGenerator. Úkolem těchto generátorů je vytvořit graf, který má vlastnosti reálných sítí. Zejména je třeba dosáhnout toho, aby rozdělení stupňů vrcholů odpovídalo reálným sítím. (5)

V rámci aplikace Measures of centrality je používán vlastní jednoduchý algoritmus pro vytvoření náhodného grafu. Tomu je předán zadaný počet vrcholů a hran a následně jsou náhodně určovány dvojice vrcholů, které jsou propojeny hranou. Vždy je třeba pouze kontrolovat, zda mezi vybranými vrcholy již spojení neexistuje.

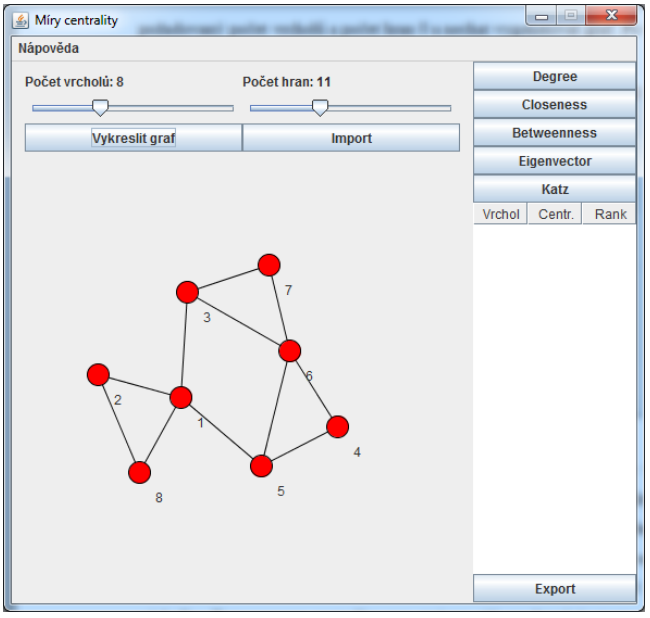
4.4. Grafické rozhraní a ovládání

Po spuštění aplikace se uživateli zobrazí základní okno (obrázek 10). Uživatel nejprve musí definovat, jaký graf si přeje analyzovat. Na dvou posuvnících má možnost zvolit si počet vrcholů a počet hran. Rozsah počtu hran se mění v závislosti na zadaném počtu vrcholů, tak aby maximum hran bylo n*(n-1)/2. K vykreslení grafu dojde po stisknutí tlačítka „Vykreslit graf“. Vrcholy grafu jsou uspořádány podle KKLayoutu, který je součástí knihovny JUNG. Umístění vrcholů je možné libovolně měnit tahem myši. U každého vrcholu je uvedeno také jeho označení, aby následně bylo jasné, ke kterému patří jednotlivé výsledky vypočítaných měr centrality.



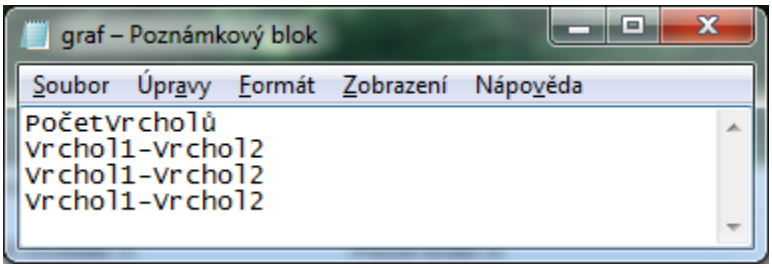
Obrázek 10: Úvodní obrazovka aplikace

Zadaný počet hran je náhodně rozmístěn mezi uzly. Jak je uvedeno i v samotné aplikaci, uživatel má následně možnost hrany grafy upravovat podle svých představ. Přidat hrany je možné kliknutím pravého tlačítka myši na vrcholu a následným tažením myši na vrchol jiný, na kterém uživatel ukončí stlačení tlačítka myši. Pokud chce uživatel nějakou hranu smazat, musí ji nejprve vybrat kliknutím levého tlačítka, hrana změní barvu, a následně stačí stisknout klávesu „delete“.



Obrázek 11: Náhodně vygenerovaný graf

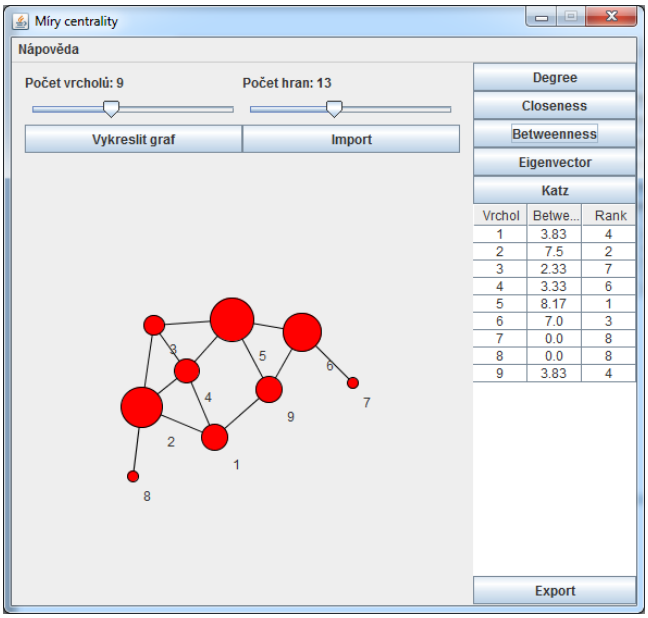
Pokud má uživatel zájem zkoumat přesně zadaný graf, může využít možnost importu grafu. K tomu je nutné zvolit textový soubor obsahující počet vrcholů a seznam všech hran. Vzor tohoto souboru je ukázán na obrázku 11.



Obrázek 12: Vzor souboru pro import grafu

Druhou možností pro analyzování přesně určeného grafu, která je vhodná, pokud uživatel nemá k dispozici seznam hran pro import, je zvolit na posuvníku jím požadovaný počet vrcholů a počet hran 0 a nechat vygenerovat graf. Po vykreslení grafu může uživatel hrany naklikat podle konkrétních představ.

Následně je již možné jednoduše kliknutím na příslušné tlačítko vyzvat aplikaci k výpočtu dané míry centrality a zobrazen výsledků. U Katz centrality ještě aplikace vyzve uživatele k zadání váhového parametru. Výsledky jsou poté zobrazeny v tabulce v pravé části okna aplikace. Tabulka obsahuje tři sloupce – název vrcholu, hodnota vypočítané centrality a pořadí vrcholu v této míře. Na obrázku 13 jsou zobrazeny výsledky Betweenness centrality vykresleného grafu.



Obrázek 13: Ukázka zobrazení výsledků

Jak je také vidět na obrázku 13, vrcholy mají různou velikost. Zvýrazněny jsou důležité vrcholy podle zvolené centrality.

Pod tabulkou s výsledky se nachází tlačítko „export“, pomocí kterého lze výsledky uložit do souboru csv. Jak vypadají exportované výsledky otevřené v Excelu, je vidět na obrázku 14. Spolu s tímto souborem dojde také k vytvoření stejnojmenného souboru s příponou txt, který

má správný formát pro import grafu a umožňuje tedy v budoucnu zobrazit v aplikaci znovu stejný graf.

	A	B	C	D	E	F
1	Vrchol	Degree	Closeness	Betweenr	Eigenvect	Katz 0.5
2	1	3.0	0.07	3.83	0.35	3.0
3	2	4.0	0.07	7.5	0.4	4.33
4	3	3.0	0.07	2.33	0.39	3.67
5	4	4.0	0.08	3.33	0.47	4.33
6	5	4.0	0.08	8.17	0.42	1.67
7	6	3.0	0.06	7.0	0.24	1.0
8	7	1.0	0.04	0.0	0.07	1.0
9	8	1.0	0.05	0.0	0.12	1.67
10	9	3.0	0.07	3.83	0.3	0.33

Obrázek 14: Exportované výsledky

4.5. Struktura aplikace

Aplikační třídy jsou rozděleny do několika balíčků (package):

- centrality
  - App – aplikační třída obsahující metodu main, která zajišťuje spuštění aplikace, tato třída také uchovává klíčovou instanci grafu,
- centrality.algorithms
  - ShortestPaths – třída obsahující metody pro výpočet nejkratších cest v grafu,
- centrality.gui
  - CentralityBox – pravý panel obsahující tlačítka a tabulku s výsledky,
  - ControlsFrame – okno zobrazující návod k ovládání aplikace,
  - Gui – třída starající se o grafické rozhraní,
  - KatzParameterFrame – okno, ve kterém se zadává váhový parametr před výpočtem KatzCentrality,
  - MenuPanel – horní panel obsahující grafické prvky pro ovládání aplikace,

- centrality.measures
    - AbstractMeasure – abstraktní míra centrality, od které dědí všech pět konkrétních měr,
    - BetweennessC – implementace IMeasure pro výpočet Betweenness centrality,
    - ClosenessC – implementace IMeasure pro výpočet Closeness centrality,
    - DegreeC – implementace IMeasure pro výpočet Degree centrality,
    - EigenvecotrC – implementace IMeasure,
    - IMeasure – interface pro míry centrality,
    - KatzC – implementace IMeasure pro výpočet Katz centrality,
- centrality.model
    - Vertex – modelová třída pro vrchol grafu, obsahuje atributy uchovávající ID, hodnoty jednotlivých měr centrality a pořadí v nich.

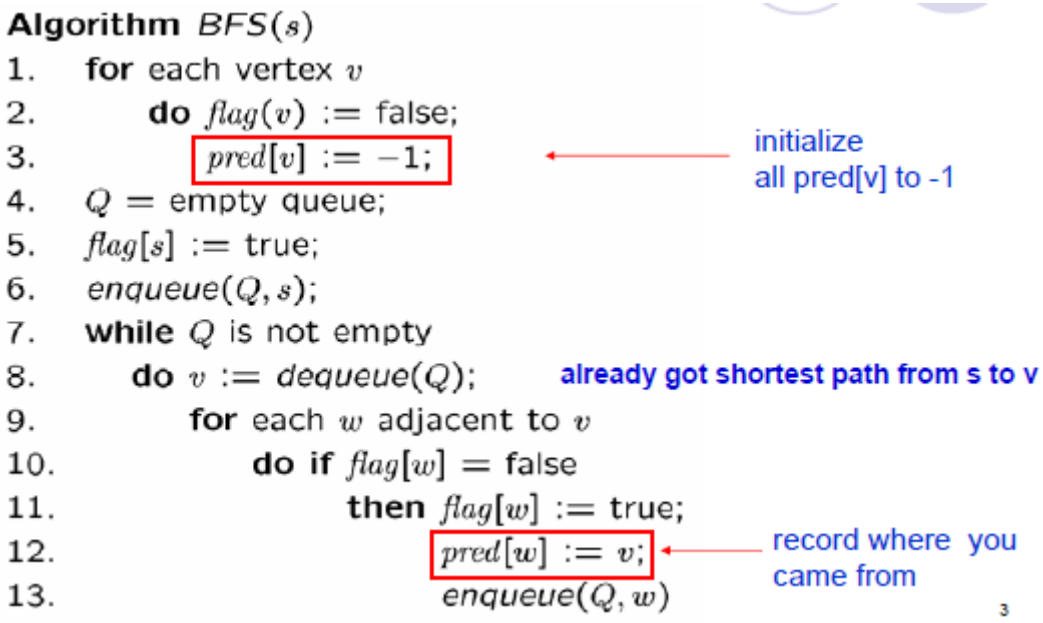
4.6. Algoritmus pro hledání nejkratších cest

Ještě před vysvětlením algoritmů pro výpočet jednotlivých měr centrality je vhodné věnovat se hledání nejkratších cest v grafu. Tyto cesty je nutné najít při výpočtu closeness a betweenness centrality.

Problematiku hledání nejkratších cest řeší v aplikaci třída ShortestPaths. Klíčová metoda této třídy getAllPaths najde všechny existující cesty mezi dvěma zadanými vrcholy grafu. Tuto metodu pak dále využívají další metody, které slouží k nalezení všech nejkratších cest (getShPaths), k nalezení všech nejkratších cest procházejících určitým vrcholem (getShPathsThru) a k určení délky nejkratší cesty (getShPathLength).

Hledání všech nejkratších cest je založeno na prohledávání grafu do šířky. Ze zdrojového vrcholu algoritmus prochází všechny ostatní vrcholy postupně od nejnižší vzdálenosti k nejvyšší. Vzdáleností je myšlen počet hran mezi těmito dvěma vrcholy. (42)

Pro nalezení nejkratší cesty je nutné při prohledávání zaznamenávat u každého vrcholu vrchol předchozí, tak aby mohla být později cesta rekonstruována.



Obrázek 15: Pseudokód pro hledání nejkratší cesty pomocí prohledávání grafu do šířky, převzato z: http://www.eecs.yorku.ca/course_archive/2006-07/W/2011/Notes/BFS_part2.pdf

Nejprve je každý vrchol grafu označený jako nezpracovaný (flag) a předchozí vrchol v prohledávání není znám, takže pred je nastaveno na -1. Prohledávání začíná ve zdrojovém vrcholu s, který je tak jako první označen jako zpracovaný a zároveň je přidán do fronty Q. Poté již může být spuštěn while cyklus, který běží, dokud fronta Q není prázdná.

Při každém průběhu cyklu je nejprve z fronty odebrán vrchol, který je přiřazen do proměnné v. Poté jsou postupně procházeny všechny jeho sousední vrcholy a v případě, že zatím nebyly zpracovány, jsou v této chvíli označeny za zpracované, jako jejich předchozí vrchol je nastaveno v a jsou přidány do fronty.

Takto jsou nalezeny nejkratší cesty mezi vrcholem s a všemi ostatními vrcholy grafu. Z libovolného vrcholu je možné postupně vypisovat předchozí vrchol procházení (pred) a dojít tak až k vrcholu s.

Pro výpočet betweenness centrality je nutné znát všechny nejkratší cesty mezi vrcholy, takže je nutné algoritmus upravit tak, aby nezaznamenával pouze jednu nejkratší cestu. Finální podobu algoritmu je možné prohlédnout si v příloze práce.

4.7. Algoritmy pro výpočet metrik

4.7.1. Degree centrality

```
for (int i = 0; i < vertices.size(); i++) {
      float centrality = (float)g.getNeighborCount(vertices.get(i).getId());
      vertices.get(i).setCentrality(centrality, 0); //0 = degree

}
```

Ukázka kódu 1: Výpočet Degree centrality

Při výpočtu Degree centrality je nutné pouze postupně procházet všechny vrcholy a zjišťovat počet jejich sousedních uzlů. To zajišťuje metoda getNeighborCount třídy Graph z knihovny JUNG, která má jako jediný parametr název příslušného vrcholu. Zjištěná hodnota se poté přiřadí uzlu jako jeho Degree centrality. Metoda setCentrality modelové třídy Vertex má kromě parametru hodnoty centrality také celočíselný parametr, který určuje, o kterou z centralit se jedná.

4.7.2. Closeness centrality

```
for (int i = 0; i < vertices.size(); i++) {
      int totalDistance = 0;
      int iVertexId = vertices.get(i).getId();
      for (int j = 0; j < vertices.size(); j++) {
            if(i!=j){
                  int jVertexId = vertices.get(j).getId();
                  if(sp.getShPathLength(iVertexId, jVertexId)!=-1){
                        totalDistance = totalDistance
                        +sp.getShPathLength(iVertexId, jVertexId);
                  }
            }
      }

      if (totalDistance > 0) {
            float n = (float) 1 / totalDistance;
            float r = (float) Math.round(n * 100) / 100;
            vertices.get(i).setCentrality(r, 1); //1 = closeness
      }
      else{
            vertices.get(i).setCentrality(0, 1); //1 = closeness
      }
}
```

Ukázka kódu 2: Výpočet Closeness centrality

Pro výpočet closeness centrality je nutné pro každý vrchol vypočítat nejkratší cestu ke všem ostatním vrcholům grafu a všechny délky sečíst. Součet těchto délek znázorňuje proměnná totalDistance, která je nejprve nastavena na 0. V cyklu jsou tedy postupně procházeny všechny vrcholy a pomocí metody getShPathLength, která byla dříve

představena, je vypočítávána délka cesta mezi vrcholem i a vrcholem j. Je nutné zajistit, aby do vrcholu j nebyl přiřazen vrchol i. Vypočítaná délka je přičtena k totalDistance.

Metoda getShPathLength vrací -1 v případě, že mezi zadanými vrcholy neexistuje žádná cesta a tyto vrcholy nejsou součástí stejné komponenty. Je tedy nutné zajistit, aby se v takovém případě získaná hodnota nepřičítala.

Po výpočtu totalDistance je již možné do proměnné n přiřadit její převrácenou hodnotu, což je hledaný výsledek. Následně dojde ještě k zaokrouhlení na 2 desetinná místa a tato hodnota již může být vrcholu přiřazena jako closeness centrality.

4.7.3. Betweenness centrality

```
for (int x = 0; x < vertices.size(); x++) {
	float betweenness = 0;
	for (int i = 0; i < vertices.size(); i++) {
		if (i != x) {
			for (int j = 0; j < i; j++) {
				if (j != x) {
					float ratio = 0;
					int iVertexId = vertices.get(i).getId();
					int jVertexId = vertices.get(j).getId();
					int xVertexId = vertices.get(x).getId();

					float ijx = sp.getShPathsThru(iVertexId,
					jVertexId, xVertexId).size();
					float ij = sp.getShPaths(iVertexId,
					jVertexId).size();

					if (ij != 0) {
						ratio = ijx / ij;
					}
					betweenness = betweenness + ratio;
				}
			}
		}
	}
	float r = (float) Math.round(betweenness * 100) / 100;
	vertices.get(x).setCentrality(r, 2); // 2 = betweenness centrality
}
```

Ukázka kódu 3: Výpočet Betweenness centrality

Betweenness centrality bodu x se vypočítává také pomocí nejkratších cest. Je nutné projít všechny varianty dvojic i, j ostatních vrcholů grafu. Mezi všemi těmito dvojicemi se vypočítá poměr počtu nejkratších cest procházejících bodem x (proměnná ijx) a počtem všem nejkratších cest (proměnná ij). Tyto poměry se sčítají v proměnné betweenness.

Nejkratší cesty procházejících daným vrcholem je možné získat pomocí metody getShPathsThru, která, stejně jako metoda getShPaths, vrací List. Počet nejkratších cest je tedy možné získat pomocí metody size.

Před nastavením betweenness centrality vrcholu x dojde ještě k zaokrouhlení na 2 desetinná místa stejně jako u všech ostatních měr.

4.7.4. Eigenvector centrality

```
private void createAdjacencyMatrix() {
	int verticesNr = g.getVertexCount();
	matrix = new double[verticesNr][verticesNr];
	for (int i = 0; i < verticesNr; i++) {
		for (int j = 0; j < verticesNr; j++) {
			if (g.isNeighbor(i + 1, j + 1)) {
				matrix[i][j] = 1;
			} else {
				matrix[i][j] = 0;
			}
		}
	}
}
```

Ukázka kódu 4: Metoda vytvářející matici sousednosti

Pro výpočet eigenvector a katz je nutné sestrojit matici sousednosti. To zajišťuje metoda createAdjacencyMatrix. Matice sousednosti je reprezentována polem matrix. Příslušné hodnoty tohoto pole náležící vrcholu i a j jsou určeny podle toho, zda jsou spolu tyto vrcholy propojeny. To zjišťuje metoda isNeighbor grafu g. Pokud spolu sousedí na příslušnou pozici je nastavena hodnota 1, v opačném případě 0.

```
private int getMaxEigenValuePos() {
	DoubleMatrix1D eigenValuesMatrix = ed.getRealEigenvalues();
	maxEigenValue = 0;
	int position = -1;
	for(int i = 0; i < eigenValuesMatrix.size(); i++){
		if(eigenValuesMatrix.get(i) > maxEigenValue){
			maxEigenValue = (float) eigenValuesMatrix.get(i);
			position = i;
		}
	}
	return position;
}
```

Ukázka kódu 5: Metoda, která vypočítá maximální vlastní číslo matice a vrátí jeho pozici

Na ukázce kódu 9 je znázorněno, jak probíhá určení pozice maximálního vlastního čísla pomocí knihovny Colt. Tuto pozici je nutné znát pro výpočet vlastního vektoru, pomocí

kterého se určuje eigenvector centralita. Všechny vlastní čísla jsou získávány pomoc metody getRealEigenvalues.

```
private void setEigenvector(int position) {
	DoubleMatrix2D eigenvectorsAll = ed.getV();
	eigenvector = new float[vertices.size()];
	for(int i = 0; i < vertices.size(); i++){
		eigenvector[i]=(float) eigenvectorsAll.get(i, position);
	}
}
```

Ukázka kódu 6: Metoda vypočítávající vlastní vektor příslušící vlastnímu číslu

Další metoda potřebná k výpočtu eigenvector centrality je setEigenvector s parametrem určujícím pozici největšího vlastního čísla získanou pomocí getMaxEigenValuePos. Metoda setEigenvector nastavuje vlastní vektor. Nejprve jsou do matice eigenvecotrsAll přiřazeny všechny vlastní vektory pomocí metody getV. Z této matice je následně získáno pole eigenvector, které obsahuje pouze hodnoty příslušící nejvyššímu vlastnímu číslu.

```
int position = getMaxEigenValuePos();
setEigenvector(position);

for (int i = 0; i < vertices.size(); i++) {
	float sum = 0;
	for(Integer n:g.getNeighbors(i+1)){
		sum = sum + eigenvector[n-1];
	}
	float centrality = 1/maxEigenValue * sum;
	float r = (float) Math.round(centrality * 100) / 100;
	if(r>0){
		vertices.get(i).setCentrality(r, 3); // 3 = Eigenvector
	}
	else{
		vertices.get(i).setCentrality(-r, 3); // 3 = Eigenvector
	}
}
```

Ukázka kódu 7: Výpočet Eigenvector centrality

Algoritmus pro výpočet eigenvector centrality nejprve zajistí nastavení příslušného vlastního vektoru pomocí metod getMaxEigenValuePos a setEigenvector. Poté se pro každý vrchol i vypočítá sum, což je součet hodnot vlastního vektoru příslušící všem sousedům vrcholu i. Výsledná hodnota eigenvector se poté vypočítá jako převrácená hodnota maximálního vlastního čísla vynásobená sum.

4.7.5. Katz centrality

```
double[] ones = new double[vertices.size()];
for (int i = 0; i < vertices.size(); i++) {
	ones[i] = 1;
}
int[][] I = new int[vertices.size()][vertices.size()];
for (int i = 0; i < vertices.size(); i++) {
	for (int j = 0; j < vertices.size(); j++) {
		if (i != j) {
			I[i][j] = 0;
		} else {
			I[i][j] = 1;
		}
	}
}

for (int j = 0; j < vertices.size(); j++) {
	for (int l = 0; l < vertices.size(); l++) {
		A[j][l] = k * A[j][l];
	}
}

for (int j = 0; j < vertices.size(); j++) {
	for (int l = 0; l < vertices.size(); l++) {
		A[j][l] = I[j][l] - A[j][l];
	}
}

DoubleMatrix2D matrix = new DenseDoubleMatrix2D(A);
Algebra alg = new Algebra();
matrix = alg.inverse(matrix);

for (int j = 0; j < vertices.size(); j++) {
	for (int l = 0; l < vertices.size(); l++) {
		matrix.set(j, l, matrix.get(j, l) - I[j][l]);
	}
}

DoubleMatrix1D onesMatrix = new DenseDoubleMatrix1D(ones);
DoubleMatrix1D centralityMatrix = alg.mult(matrix, onesMatrix);

if(centralityMatrix.get(0)<0){
	for(int i = 0; i<centralityMatrix.size();i++){
		centralityMatrix.set(i, centralityMatrix.get(i)*-1);
	}
}

for (int i = 0; i < centralityMatrix.size(); i++) {
	float c = (float) centralityMatrix.get(i);
	float r = (float) Math.round(c * 100) / 100;
	vertices.get(i).setCentrality(r, 4);

}
```

Ukázka kódu 8: Výpočet Katz centrality

Pro správný výpočet katz centrality je nutné správně vynásobit matice podle vzorce $KC(x) = 1^T((I_n - kA)^{-1} - I_n)e_x$. Známá je matice sousednosti A a váhový faktor k. Dále je nutné vytvořit vektor ones, který má na všech pozicích hodnotu 1 a jednotkovou matici I.

Nejprve se všechny prvky matice A vynásobí koeficientem k. Dále je nutné vypočítat rozdíl matic I a A. Z tohoto výpočtu se určí inverzní matice. Od získané inverzní matice je nutné dále odečíst matici I. Zbývá již jen výslednou matici vynásobit vektorem ones a je znám výsledný vektor centralityMatrix, který obsahuje hodnoty centrality pro všechny vrcholy grafu.

Katz centrality vrcholu i je tak rovna hodnotě pole centralityMatrix na pozici i. Zbývá tuto hodnotu zaokrouhlit a přiřadit vrcholu pomocí setCentrality.

4.8. Ukázkový příklad

4.8.1. Zadání

Jako ukázkový příklad řešený vytvořenou aplikací Measures of centrality byla zvolena analýza centralit aktérů kolaborační sítě oblíbených filmových herců a hereček.

V kolaboračních sítích jsou spojení mezi uzly ustanovována pomocí členství ve skupinách. Často je zkoumána spolupráce vědců, která je posuzována podle spoluautorství odborných prací. V síti spolupracujících herců jsou propojeni aktéři, kteří hráli alespoň v jednom filmu společně. (1)

Do sítě je zahrnuto 6 nejoblíbenějších žen a 6 mužů podle Česko-Slovenské filmové databáze (csfd.cz), kde je oblíbenost posuzována podle počtu uživatelů, kteří se označí za fanouška daného herce/herečky. Existují i rozsáhle analýzy sítí obsahujících všechny hollywoodské herce, tento příklad má však pouze ukázat způsob práce s vytvořenou aplikací a správnou reprezentaci výsledků, a tak bude názornější v menší síti obsahující pouze 12 vrcholů.

Přítomnost vazby mezi jednotlivými herci je zjišťována podle Internetové filmové databáze (imdb.com), která poskytuje výpis všech společných děl dvou zadaných osob. Herci jsou propojeni pouze v případě, že spolu hráli ve filmu. Existenci vazby nezpůsobí společná přítomnost v seriálech, ve kterých se často oblíbení herci objeví pouze jako hosté jedné epizody, nebo například záznamy z různých ocenění, které jsou také evidovány v Internetové filmové databázi. Pokud se dvě osoby podílejí na filmu,

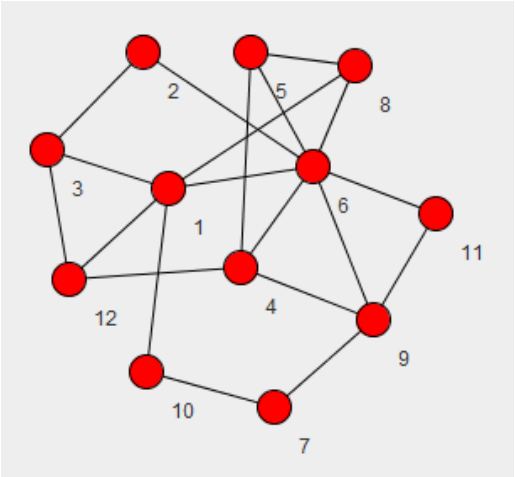
ale jedna z nich v jiné než herecké pozici, například jako producent, tak v síti také nejsou propojeni.

Jednotliví herci jsou reprezentováni vrcholy označenými čísly. Označení herců je následující:

- 1 - Johny Depp,
- 2 - Tom Hanks,
- 3 - Leonardo Di Caprio,
- 4 - Bruce Willis,
- 5 - Brad Pitt,
- 6 - Morgan Freeman,
- 7 - Natalie Portman,
- 8 - Angelina Jolie,
- 9 - Scarlett Johansson,
- 10 - Keira Knightley,
- 11 - Jennifer Aniston,
- 12 - Maryl Streep.

4.8.2. Výpočet

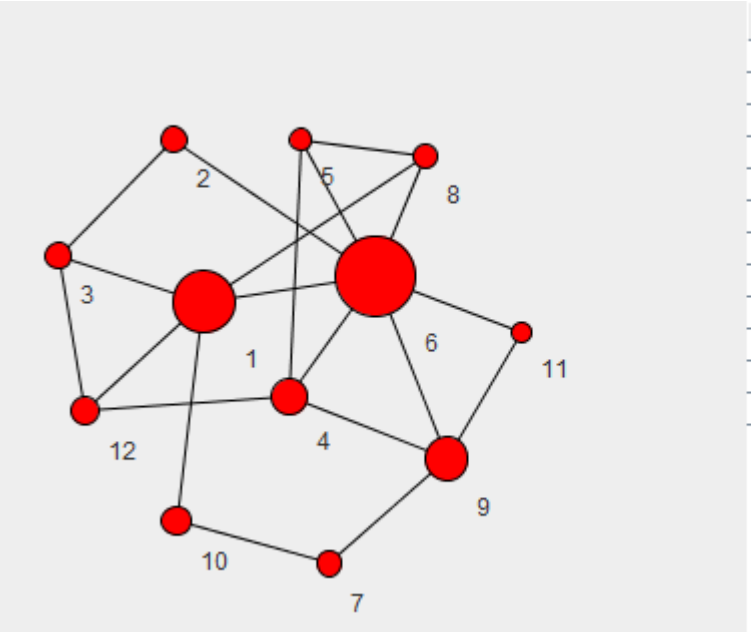
Pro vložení sítě ke zpracování do aplikace je vhodné využít možnosti importování sítě. Je tak nutné vytvořit textový soubor obsahující všechny hrany. Jak vypadá síť vykreslená v aplikaci Measures of centrality, je možné vidět na obrázku 16. Díky layoutu je již z obrázku možné vyčíst, že mezi důležitější vrcholy sítě by měly patřit vrcholy 1, 4 a 6 umístěné uprostřed grafu.



Obrázek 16: Vykreslená síť kolaborujících herců

4.8.3. Výsledky

Po vložení sítě do aplikace je již možné zobrazit výsledky jednotlivých centralit. Na obrázku 17 je vidět, jaké jsou výsledky betweenness centrality. Velikost vrcholů odpovídá výsledku centrality, takže je zřejmé, že kromě tří vrcholů umístěných uprostřed grafu je významný také vrchol 9, což potvrzuje také výsledek betweenness centrality uvedený v pravém sloupci.



Vrchol	Bet...	Rank
6	19.67	1
1	13.83	2
9	7.67	3
4	5.25	4
10	3.0	5
12	2.42	6
3	2.0	7
7	1.83	8
2	1.58	9
8	1.25	10
5	0.5	11
11	0.0	12

Obrázek 17: Zobrazené výsledky betweenness centrality

Pro přehledný výpis výsledků všech pěti hodnot centralit je možné použít možnost exportu výsledků. Tabulka 10 obsahuje hodnoty zkopírované z exportovaného csv souboru.

Vrchol	Degree	Closeness	Betweenness	Eigenvector	Katz 0,5
1 – Depp	5	0,06	13,83	0,35	53
2 – Hanks	2	0,05	1,58	0,19	17
3 – Di Caprio	3	0,04	2	0,2	59
4 - Willis	4	0,05	5,25	0,35	25
5 - Pitt	3	0,05	0,5	0,3	23
6 - Freeman	7	0,07	19,67	0,52	27
7 - Portman	2	0,04	1,83	0,11	13
8 - Jolie	3	0,05	1,25	0,3	3
9 – Johansson	4	0,05	7,67	0,31	49
10 - Knightley	2	0,04	3	0,12	21
11 - Aniston	2	0,04	0	0,21	37
12 - Streep	3	0,05	2,42	0,23	45

Tabulka 12: Exportované hodnoty měr centrality

Hodnota degree uvádí, s kolika herci v rámci sítě se příslušný aktér objevil ve filmu. Z tohoto hlediska je jednoznačně nejaktivnější Morgan Freeman, který se setkal se 7 z 11 nejoblíbenějších herců podle csfd. Druhou nejvyšší hodnotu degree centrality má Johny Depp, který se setkal s 5 herci z této sítě. S nejmenším počtem 2 herců spolupracovali Tom Hanks, Natalie Portman, Keira Knightley a Jennifer Aniston.

Closeness centrality v tomto případě vychází podobně jako degree centrality. Za předpokladu, že se znají pouze herci, kteří spolu hráli v nějakém filmu a zprávy se dají posílat pouze prostřednictvím herců z této sítě vždy pouze mezi dvěma, kteří se znají, by měl nejlepší pozici Morgan Freeman, který má nejnižší vzdálenost k ostatním aktérům sítě a potřeboval by nejnižší počet zprostředkovatelů k doručení zpráv všem ostatním.

Pokud platí stejné předpoklady, které byly uvedeny u reprezentace výsledků closeness centrality, tak lze výsledky betweenness centrality reprezentovat tímto způsobem: největší kontrolu nad informacemi v této síti má Morgan Freeman, který leží na vysokém počtu nejkratších cest mezi ostatními aktéry. Jako nejméně vhodný prostředník pro šíření informací je vyhodnocena Jennifer Aniston, u které výsledek betweenness 0 ukazuje, že je zcela nedůležitá pro propojení ostatních herců.

Výsledné hodnoty eigenvector centrality ukazují, že nejvíce propojené sousedy má Morgan Freeman. Na druhém místě se v tomto případě spolu s Johny Deppem nachází Bruce Willis, který má sice méně přímých spojení, ale s více propojenými sousedy, takže u obou těchto herců vychází eigenvector centrality 0,35. Jako nejméně důležití aktéři z hlediska eigenvector centrality se ukazují Keira Knightley s výsledkem 0,12 a Natalie Portman s výsledkem 0,11.

Katz centralita byla vypočítána s hodnotou váhového faktoru 0.5. V případě Katz centrality vychází jako nejdůležitější herec pro šíření informací v síti Leonardo Di Caprio, který má tak zřejmě dobrá nepřímá spojení. V důležitosti aktérů sítě podle Katze je na druhém místě Johny Depp. Nejnižší hodnotu Katz centrlity má Angelina Jolie.

U tohoto příkladu vychází jako nejdůležitější shodně ve čtyřech z pěti měr centrality Morgan Freeman. Nejdůležitější aktér sítě je rozdílný pouze v centralitě podle Katze, která i u ostatních aktérů nabízí nejvíce rozdílná řešení v porovnání s ostatními měrami.

I v dalším pořadí dochází ke změnám podle přístupu k výpočtům jednotlivých měr. Například Brad Pitt je v pořadí eigenvector centrality na 5. místě, ale hodnotu betweenness centrality má 2. nejnižší. Tento rozdíl je možné vysvětlit tím, že má poměrně dobře propojené sousedy, což je důležitě z hlediska eigenvector centrality, ale leží na malém počtu nejkratších cest mezi ostatními vrcholy, což je zásadní z hlediska betweenness centrality.

Další rozdíl v pořadí mezi měrami je vidět například mezi výsledkem degree centrality a closeness centrality u Toma Hankse a Leonarda Di Capria. Degree centrality má Di Caprio vyšší než Hanks, ale u Closeness centrality je toto pořadí opačné. Toto ukazuje, že u výpočtu closeness centrality nerozhoduje pouze počet sousedních vrcholů jako u degree centrality, ale délka cest ke všem ostatním vrcholům sítě, která je v případě Hankse nižší než u Di Capria, přestože má Hanks méně sousedů.

4.9. Testování aplikace

Aplikace byla testována na Windows XP, Windows 7 32-bit i 64-bit a Windows 8. Na všech těchto operačních systémech byly ověřeny všechny její funkce. Podařilo se dosáhnout toho, že se aplikace ve všech uvedených případech chovala podle předpokladů a vypočítávala očekávané a ověřené výsledky a mimo jiné je zvládala i úspěšně exportovat do souboru csv.

5. Závěr

Tato práce se věnuje analýze sociálních sítí. Hlavním úkolem práce je popsat a porovnat metriky pro určení míry centrality aktérů sociálních sítí.

Cílem práce je seznámení se základními teoretickými znalostmi týkajícími se komplexních sítí, zejména o jejich analýze s důrazem na centralitu vrcholů sítě, a dále popis a implementace několika základních algoritmů měr centrality včetně jejich optimalizačních metod.

Teoretická část obsahuje seznámení s komplexními sítěmi a popisuje, jak jsou rozděleny podle toho, jakou situaci reálného světa zachycují. Dále je čtenář seznámen s možnými přístupy k analýze sítí včetně několika základních statistik, pomocí kterých jsou sítě popisovány. Poté je již ve zbytku teoretické části pozornost věnována hlavnímu tématu, kterým jsou metriky pro určení centrality sítí.

V další části práce jsou nejprve vysvětleny vzorce pro výpočet jednotlivých centralit a všechny jsou aplikovány v jednoduchém ukázkovém příkladu. U všech metrik je také věnována pozornost možnostem optimalizace výpočtu. Hlavním úkolem praktické části je vytvoření aplikace, která umožňuje zadání grafu sítě a následně vypočítá pět základních metrik míry centrality aktérů této sítě. Pro vytvoření aplikace byl zvolen programovací jazyk Java. V rámci popisu vývoje aplikace jsou uvedeny a vysvětleny algoritmy, které jsou požity pro výpočet centrality. Na závěr je aplikace použita k provedení analýzy malé sítě reálného světa. Jedná se o kolaborační síť dvanácti nejoblíbenějších filmových herců.

Tato práce poskytuje přehledný a podrobný rozbor základních metrik pro určení centrality. Podařilo se vytvořit aplikaci, která všechny tyto metriky dokáže vypočítat pro libovolnou, uživatelem zadanou, síť. Práce může posloužit především jako studijní materiál pro každého, koho zajímá zejména centralita sítí.

Za nedostatek práce může být považováno, že se věnuje pouze pěti metrikám centrality, i když jde o ty, které jsou používány ve většině případů. Dalším nedostatkem je, že algoritmy implementované v aplikaci nejsou optimalizované. Aplikace posloužila především k vyzkoušení a vysvětlení algoritmů, jsou dostupné jiné a více propracované aplikace, které se věnují analýze sítí. Patří mezi ně například aplikace Gephi.

Tato práce byla zpracována s pomocí několika knih a vědeckých článků. Většina zdrojů byla nalezena v databázích odborné literatury Scopus, Springer a Web of Science, ke kterým umožňuje přístup Univerzita Hradec Králové. Některé další vědecké publikace byly nalezeny také s pomocí vyhledávače Google Scholar.

6. Zdroje

1. **Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., Hwang, D.U.** Complex networks: Structure and dynamics. *Physics Reports.* 2006, Sv. 424, stránky 175-308.

2. **Newman, M.E.J.** The structure and function of complex networks. *Siam Review.* 2003, Sv. 45, stránky 167-256.

3. **Jackson, M.O.** *Social and Economic Networks.* Princeton, NJ : Princeton University Press, 2008. ISBN: 9780691148205.

4. **Matoušek, J., Nešetřil, J.** *Invitation to Discrete Mathematics.* Oxford : Oxford University Press, 2008. ISBN: 9780198570431.

5. **Albert, R., Barabasi, A.L.** Statistical mechanics of complex networks. *Reviews of Modern Physics.* 2002, Sv. 74, stránky 47-97.

6. **Barabási, A.L., Albert, R., Jeong, H.** Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A.* 2000, Sv. 281, stránky 69-77.

7. **Travers, J., Milgram, S.** An Experimental Study of the Small World Problem. *Sociometry.* 1969, Sv. 32, stránky 425-443.

8. **Zhang, P.P., Chen K., He, Y., Zhou, T., Su, B.B., Jin, Y., Chang, H., Zhou, Y.P., Sun, L.C., Wang, B.H., He, D.R.** Model and empirical study on some collaboration networks. *Physica A: Statistical Mechanics and its Applications.* 2006, Sv. 360, stránky 599-616.

9. **Landherr, A., Friedl, B., Heidemann, J.** A Critical Review of Centrality Measures in Social Networks. *Business & Information Systems Engineering.* 2010, Sv. 2, stránky 371-385.

10. **Chojnacki, S., Ciesielski, K., Klopotek, M.** Node Degree Distribution in Affiliation Graphs for Social Network Density Modeling. *Social Informatics.* 2010, Sv. 6430, stránky 51-61.

11. **Barabási, A.L., Jeong, H, Néda, Z., Ravasz, E., Schubert, A., Vicsek, T.** Evolution of the social network of scientific collaborations. *Physica A.* 2002, Sv. 311, stránky 519-614.

12. **Luo, X. Yu, J.X., Li, Z.** *Advanced Data Mining and Applications.* Hangzhou : Springer, 2014. ISBN: 978-3-642-35526-4.

13. **Glass, K., Colbaugh, R., Ormerod, P., Tsao, J.** *Complex Sciences.* Berlín : Springer Berlin, 2013. ISBN: 978-3-319-03472-0.

14. **Aggarwal, C.C.** *Social Network Data Analytics.* místo neznámé : Springer Publishing Company, 2011. ISBN: 978-1-4419-8461-6.

15. **Alhajj, R., Rokne, J.** *Encyclopedia of Social Network Analysis and Mining.* New York : Springer New York, 2014. ISBN: 9781461461708.

16. **Wolf, F., Mohr, B., Mey, D.** *Euro-Par 2013 Parallel Processing.* Berlín : Springer Berlin Heidelberg, 2013. ISBN: 978-3-642-40046-9.

17. **Brautbar, M., Kearns, M.** A Clustering Coefficient Network Formation Game. *Lecture Notes in Computer Science.* 2011, Sv. 6982, stránky 224-235.

18. **Liu, Z., Wang, C., Zou, Q., Wang, H.** Clustering Coefficient Queries on Massive Dynamic Social Networks. *Lecture Notes in Computer Science.* 2010, Sv. 6184, stránky 115-126.

19. **Wasserman, S., Faust, K.** *Social Network Analysis: Methods and Applications.* Cambridge : Cambridge University Press, 1994. ISBN 0-521-38269-6.

20. **Borgatti, S.P.** Centrality and network flow. *Social Networks.* 2005, Sv. 27, stránky 55-71.

21. **Brandes, U., Erlebach, T.** *Network Analysis.* Berlin : Springer Berlin Heidelberg, 2005. ISBN: 978-3-540-24979-5.

22. **Bonacich, P.** Some unique properties of eigenvector centrality. *Social Networks.* 2007, Sv. 29, stránky 555-564.

23. **Perra, N., Fortunato, S.** Spectral centrality measures in complex networks. *Physical Review E.* 2008, Sv. 78.

24. **Brin, S., Page, L.** The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems.* 1998, Sv. 30, stránky 107-117.

25. **Koschützki, D., Lehmann, K. A., Peeters, L., Richter, S., Tenfelde-Podehl, D., Zlotkowski, O.** Centrality Indices. *Network Analysis.* 2005, Sv. 3418, stránky 16-61.

26. *Parallel Algorithms for Evaluating Centrality Indices in Real-World Networks.* **Bader, D.A., Madduri, K.** Columbus, OH : Georgia Institute of Technology, 2006. stránky 539-550. ISBN: 0-7695-2636-5.

27. **Khopkar, S.S., Nagi, R., Nikolaev, A.G., Bhembre, V.** Efficient algorithms for incremental all pairs shortest paths, closeness and betweenness in social network analysis. *Social Network Analysis.* 2014, Sv. 4, stránky 1-20.

28. **Okamoto, K., Chen, W., Li, X.-Y.** Ranking of Closeness Centrality for Large-Scale Social Networks. *Frontiers in Algorithmics.* 2008, Sv. 5059, stránky 186-195.

29. **Kourtellis, N., Alahakoon, T., Simha, R., Iamnitchi, A., Tripathi, R.** Identifying high betweenness centrality nodes in large social networks. *Social Network Analysis.* 2013, Sv. 3, stránky 899-914.

30. **Green, O., Bader, D.A.** Faster Betweenness Centrality Based on Data Structure Experimentation. *Procedia Computer Science.* 2013, Sv. 18, stránky 399-408.

31. **TeamFoxes.** Tutorial on Numerical Analysis with Matrix.xla. [Online] 2006. http://ue.poznan.pl/data/upload/articles/20140213/4e8165990155700259/matrixtutorial2.pdf.

32. **Woodford, C., Phillips, C.** *Numerical Methods with Worked Examples: Matlab Edition.* 2. místo neznámé : Springer Netherlands, 2012. ISBN: 978-94-007-1365-9.

33. **Sharma, S., Purohit, G.N.** Evaluation of Alternative Centrality Measure Algorithm For Tracking Online. *International Journal of Engineering Research.* 2012, Sv. 1, stránky 28-33.

34. **Foster, K.C., Muth, S.Q.** A Faster Katz Status Score Algorithm. *Computational & Mathematical Organization Theory.* 2001, Sv. 7, stránky 275-285.

35. **Sander, G.** Graph layout for applications in compiler construction. *Theoretical Computer Science.* 1999, Sv. 217, stránky 175-214.

36. **Gibson, H., Faith, J., Vickers, P.** A survey of two-dimesional graph layout techniques for information visualisation. *A survey of two-dimensional graph.* 2013, Sv. 12, stránky 324-357.

37. **Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.** Algorithms for Drawing Graphs: an Annotated Bibliography. *Computational Geometry: Theory and Applications.* 1994, Sv. 4, stránky 235-282.

38. **Fruchterman, T.M.J., Reingold, E.M.** Graph Drawing by Force-directed Placement. *SOFTWARE-PRACTIC AND EXPERIENCE.* 1991, Sv. 21, stránky 1129-1164.

39. **Kohonen, T.** The self-organizing map. *Self-Organizing Graphs: A Neural Network Perspective of Graph Layout.* 1990, Sv. 78, stránky 1464-1480.

40. **B., Meyer.** Self-Organizing Graphs: A Neural Network Perspective of Graph Layout. *Graph Drawing.* 1998, Sv. 1547, stránky 246-262.

41. **Kamada, T., Kawai, S.** An Algorithm for Drawing General Undirected Graphs. *Information Processing Letters.* 1989, Sv. 31, stránky 7-15.

42. **Xu, J.J., Chen, H.** Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks. *Decision Support Systems.* 2003, Sv. 38, stránky 473-487.

Přílohy

Obsah přiloženého CD:

- elektronická verze diplomové práce ve formátu pdf,
- instalační soubor aplikace Míry centrality ve formátu jar,
- zdrojové kódy aplikace,
- příklady řešené v rámci práce.

Zadání k závěrečné práci

Jméno a příjmení studenta: **Jiří Mikeš**

Obor studia: Aplikovaná informatika
Jméno a příjmení vedoucího práce: **Jiří Haviger**

Název práce:
Míry centrality v sociálních sítích

Název práce v AJ:
Measures of centrality in social networks

Podtitul práce:

Podtitul práce v AJ:

Cíl práce: Cílem práce je popsat a porovnat metriky pro určení míry centrality aktérů sociálních sítí. Náplní práce je také vysvětlení algoritmů výpočtu jednotlivých metrik a jejich optimalizace.

Osnova práce:
1. Úvod
2. Přehled metrik
3. Způsoby výpočtu
4. Algoritmy výpočtu
5. Závěr

Projednáno dne:

Podpis studenta Podpis vedoucího práce