Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Měření významnosti autorů v citační síti

Plzeň 2013 Tomáš Maršálek

Prohlášení

Abstract

Obsah

1 Úvod

V době, kdy je celý svět propojen internetem...

Přirozeně vyvstávají otázky. Kdo je nejvýznamnější člověk mezi svými přáteli? Které město je klíčovým dopravním uzlem v železniční síti? Představme si, že neznáme jednotlivé lidi v síti a žádné informace o nich, jak zjistíme, kdo je nejvýznamnější pouze na základě vztahů mezi nimi?

Tato práce je věnována citačním sítím, kde používáme metody z analýzy sociálních, dopravních, komunikačních a jiných sítí. V citační síti hledáme nejvýznamnější autory pouze podle toho, jak jsou provázáni s jinými autory podle referencí v publikacích, které vydali. Významnost autorů není úplnou neznámou, protože existuje množství ocenění, které byly uděleny právě významným autorům a vědcům za jejich dílo. Samotné udělení ocenění mohlo tyto autory udělat významnými, přestože předtím nebyli. V jiném případě mohlo být důležité ocenění příčinou ještě větší významnosti autora.

Cílem této práce je vytvořit knihovnu pro analýzu citační sítě z hlediska významnosti autorů a použít ji k porovnání jednotlivých implementovaných metod pro měření významnosti mezi sebou a k porovnání těchto metod s již udělenými oceněními. Očekáváme, že pokud měříme významnost autora nebo obecně prvku v síti, přestože neznáme její přesnou definici, bude se shodovat s těmito oceněními.

V průběhu této práce budou přiblíženy detaily o nejpoužívanějších mírách centrality (degree, eigenvector centrality, betweenness centrality, closeness centrality), bibliografické metodě h-index a jejich implementaci. Výsledkem budou porovnání jednotlivých metod aplikovaných na citační sítě vytvořené z citačních databází DBLP a CiteSeer, které zároveň srovnáme s oceněními autorů v oblasti informatiky (ACM SIGMOD Edgar F. Codd Innovations Award, ACM Fellows, ACM A.M. Turing Award, ISI Highly Cited highlighted).

2 Sociální a citační sítě

Myšlenka sociální sítě existovala dlouho předtím, než je pod tímto termínem začali lidé rozpoznávat. Jedná se o komplexní struktury vztahů mezi členy sociálních uspořádání na všech úrovních - od osobních až po mezinárodní vztahy mezi organizacemi.

Nejčastěji se ale setkáme se sociální sítí jako strukturou tvořenou lidmi, kteří jsou svázáni nějakým sociálním vztahem, zejména v poslední době s rozmachem populárních webových sociálních sítí (MySpace, Facebook, G+, Lidé), jím bývá přátelství.

2.1 Reprezentace sítí

Abychom mohli pracovat s doposud abstraktním konceptem sítě, musíme být schopni ji reprezentovat jako datovou strukturu, na níž poté provedeme jakoukoliv analýzu. V odvětví matematiky teorie grafů je síť (graf) dvojice množin uzlů V (vrcholů) a spojení uzlů E (hran) $G = (V, E)$. Obecně můžeme uvažovat grafy s hranami s orientací či bez orientace. V obou případech se stále jedná o dvojici (V, E), pouze pro orientovaný graf je množina hran množinou uspořádaných dvojic oproti množině neuspořádaných dvojic u neorientovaného grafu.

V definici grafu je množina hran E soubor dvojic, které označují koncové uzly hrany, neboli jejich spojení. Samotné spojení je jediná informace, kterou množina hran nese. Chceme-li zaznamenat nějakou další informaci, která je spojená se spojením dvou uzlů, namísto hrany jako dvojice koncových uzlů, nadefinujeme hranu jako n-tici, kde první dvě hodnoty jsou koncové uzly a zbylé hodnoty nesou libovolnou informaci. Ve většině případů si vystačíme s jednou dodatečnou informací a nazýváme ji váha hrany. Jiná možnost pro zavedení vah hran je váhová funkce $f : E \mapsto \mathbb{R}$, kde $f(e) = w$ je ohodnocení konkrétní hrany $e \in E$. V případě zavedení vah hovoříme o vážených sítích.

Při zavedení vah máme například možnost používat síť jako multigraf, tedy graf, u kterého je povoleno více spojení mezi dvěma stejnými uzly. Počet stejných hran pak pouze zaznamenáme celočíselnou hodnotou ve váze hrany.

Síť world wide web tvořená webovými stránkami je příkladem multigrafu, protože je povoleno z jedné stránky odkazovat na jinou na více místech. Při analýze takovýchto sítí využijeme právě vah hran a počet hypertextových odkazů mezi dvěma stránkami zaznamenáme vyšším ohodnocením hrany. V tomhle případě znamená vyšší váha silnější pouto mezi uzly.

Jiným případem může být např. síť, kde sledujeme města a dopravní

spojení mezi nimi. V tomto případě nás může zajímat vzdálenost nebo časová náročnost na dopravu mezi dvěma městy, které budou znamenat silnější pouto, pokud budou mít naopak menší váhu. Hledáme totiž nejkratší či nejrychlejší spojení.

Pro reprezentaci v paměti počítače se nejčastěji používají dva způsoby - matice sousednosti a graf pomocí spojových seznamů. Hrany se uzlu v případě orientovaného grafu liší z pohledu jednoho uzlu. Pokud hrana vychází z tohoto uzlu, nazveme ji výstupní hrana, v opačném případě se bude jednat o vstupní hranu.

Matice sousednosti (adjacency matrix) je čtvercová matice A o velikosti počtu vrcholů grafu $|V|$, ve které prvek $\mathbf{A}_{uv}$ na řádku u a sloupci v určuje jestli existuje hrana od vrcholu u do vrcholu v. Pokud je hodnota $\mathbf{A}_{uv}$ 1, hrana existuje; pokud je hodnota 0, pak hrana neexistuje a pokud je hodnota w, pak hrana existuje s váhou w.

Jiným maticovým způsobem uchování grafu je incidenční matice $\mathbf{B}$. Incidenční matice vyjadřuje vztah mezi vrcholy a hranami tak, že $\mathbf{B}_{ue} = 1$, pokud vrchol u je spojený s hranou e, a 0 v opačném případě. V orientovaném grafu rozlišujeme mezi počátečním uzlem $\mathbf{B}_{ue} = -1$ a koncovým uzlem $\mathbf{B}_{ue} = 1$. Incidenční matice se pro výpočetní teorii grafů často nepoužívá z důvodu paměťové náročnosti, která je pro většinu grafů výrazně vyšší než u matice sousednosti ($\Theta(|V||E|)$ oproti $\Theta(|V|^2)$, kde množina hran dosahuje velikostí $O(|V|^2)$).

Nejčastěji používáme myšlenku sousednosti vrcholů, ale namísto reprezentace maticí, která je ve většině případů řídká a zbytečně obsahuje velké množství nul, použijeme reprezentaci řídké matice - řádek nahradíme seznamem vrcholů, které v matici sousednosti mají nenulovou hodnotu. Tento způsob je známý jako graf pomocí spojových seznamů (adjacency list representation of a graph).

2.2 Citační sítě

Citační sítě jsou podobné sociálním sítím, pouze místo uzlů, které představují osoby, se v citační síti jedná o publikace nebo autory těchto publikací. Pokud je uzlem publikace, pak hrany této sítě symbolizují citaci publikace jinou publikací. V druhém případě uvažujeme síť, kde uzly reprezentují autory knih, vědeckých článků, vědecké literatury a dalších publikací. Prvnímu typu říkáme síť publikací, druhému síť autorů.

2.2.1 Citační síť publikací

Uvažujeme-li první případ, kde uzly reprezentují publikace a hrany přímo citace mezi těmito publikacemi, jedná se o síť publikací. Pokud publikace A odkazuje na publikaci B, budou existovat stejnojmenné uzly A a B a hrana mezi těmito uzly může mít dvě různé orientace podle svého uplatnění. Směr od citující publikace k citované (v našem příkladě od A do B) bude mít hrana, kterou označíme jako výstupní pro uzel A a vstupní pro uzel B. Výstupní hrana, laicky řečeno, označuje vztah „cituji“, kdežto vstupní hrana znamená „jsem citován“.

2.2.2 Citační síť autorů

Druhou citační sítí je citační síť autorů. Zde je uzel reprezentací autora a hrany spojují autory mezi sebou. Ve většině případech máme k dispozici data ve formátu, který přímo odpovídá síti publikací, tj. pro každou publikaci známe seznam jejích autorů a odkazů na další publikace. Síť autorů lze získat transformací sítě publikací tak, že každou hranu z původní sítě publikací přiřadíme každému z autorů citující publikace a duplikujeme ji pro každého z autorů citované publikace. Celkově vznikne nm nových hran, pokud odkazovaná publikace obsahuje n autorů a odkazující m autorů. Stejně jako v síti publikací, i zde uvažujeme opačné orientace hrany.

V síti autorů má pro naše účely smysl uvažovat ohodnocení hran. Existuje více způsobů, jak lze přiřadit ohodnocení (váhu) jednotlivým hranám, ale nejjednodušším způsobem, který je použitý i v implementaci knihovny, je prosté přiřazení počtu publikací, jejichž autorem je daný autor A, které odkazují na publikace, jejichž autorem je autor B. Srozumitelnější popis poskytne obrázek:

Druhou možností ohodnocení hran, který rovněž využívá implementovaná knihovna pro některé metody, je převrácená hodnota prvního způsobu ohodnocení. Důvodem je přímá souvislost mezi vahou hrany a vzdáleností mezi uzly. V prvním případě, kdy silnější pouto mezi autory vyjadřuje vyšší ohodnocení hrany, v druhém případě je naopak nižší váha vyjádřením silnějšího vztahu, jelikož jsou si uzly blíže. Tento způsob je používán pro algoritmy, které pracují na myšlence nejkratších cest mezi uzly.

2.2.3 Vážené citační sítě

Pro citační sítě můžeme uvažovat ohodnocení hran obojího typu. Například mezi dvěma autory může být silnější vztah, pokud se citují ve více publikacích. Jestliže citační síť analyzujeme metodami, které jsou založené na myšlence

hledání nejkratších cest i v této síti, která nemá v podstatě žádný pojem vzdálenosti, použijeme druhý typ ohodnocení - menší váha, silnější pouto.

2.2.4 Orientované a neorientované sítě

V případě sociálních sítí nejčastěji uvažujeme sítě bez orientace, protože nejčastěji modelovaný vztah přítel-přítel je ekvivalentní z pohledu obou koncových uzlů. Pro citační síť jsou na místě orientované hrany, protože vztahy autor odkazujícího na jiného autora nebo publikace citující jinou publikaci mají očividně jinou interpretaci z pohledu koncových uzlů. Buď to se jedná o citovaného nebo citujícího autora či publikaci.

3 Citační databáze

Bibliografická citační databáze poskytují možnost vyhledávání bibliografických citací. Velké množství z dnešních citačních databází se zaměřuje na jeden obor. (cite http://library.amnh.org/research-tools/citation-full-text-databases). Jiné jsou multioborové s možností volby prohledávaného oboru (Scopus, Web of Science).

3.0.5 DBLP

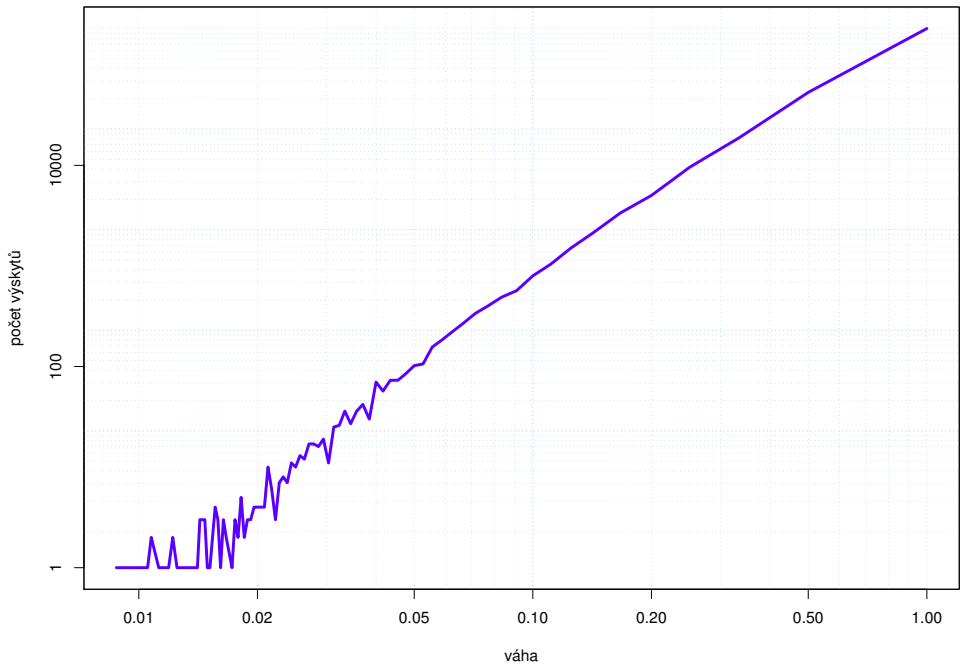
DBLP [DBL] je webová bibliografická databáze hostovaná na univerzitě Trier. Od 80. let indexovala literaturu z oblasti databázových systémů a logického programovaní, ale postupně se její zaměření zobecnilo a nyní je bibliografickou databází obecně pro obor informatiky. V roce 2012 obsahovala více než 2,1 milionu článků. Metody implementované v této práci jsou aplikovány na verzi z roku 2004.

	hodnota
Počet publikací	470 554
Počet hran v síti publikací	109 130
Počet autorů	315 485
Počet hran v síti autorů	331 245
Počet samocitací	3 095
Průměrný počet spoluautorů	2.278
Největší silně spojitá komponenta	1.637%
Průměrná nejkratší cesta[1]	2.894
Průměrná nejkratší cesta[2]	1.194
Poloměr grafu[1]	8.0
Poloměr grafu[2]	6.637

Tabulka 3.1: Statistiky pro databázi DBLP 2004

3.0.6 CiteSeer

CiteSeer (nyní CiteSeer^X) [cit] je považován za první automatizovaný systém shromažďování publikací a autonomní indexace citací v nich obsažených. Publikace jsou zejména z oboru informatiky a informační vědy. V dnešní



Obrázek 3.1: Mocninné rozdělení vah hran ($\frac{1}{w}$) citační sítě autorů DBLP

době obsahuje přes dva miliony dokumentů s téměř dvěma miliony autorů a čtyřiceti miliony citací. Zde používáme verzi z roku 2005.

[1]Platí pro neváženou síť autorů
[2]Platí pro váženou síť autorů

3.1 Analýza sítí

Více než 40 let byly považovány všechny komplexní sítě za naprosto náhodné. Paul Erdős a Alfréd Rényi v roce 1959 navrhli modelování komunikačních sítí a sítí, které se vyskytují v přírodních vědách, spojením uzlů náhodnými hranami. Tato jednoduchá metoda vytvoření náhodného grafu způsobí rozložení stupňů vrcholů (počet spojení vrcholu s ostatními) podle Poissonova rozdělení s charakteristickou křivkou připomínající zvon - většina uzlů má zhruba stejný stupeň. V roce 1998 bylo na univerzitě v Notre Dame (Barabási a kolegové) provedeno mapování sítě World Wide Web s očekáváním, že výsledkem bude náhodná síť. Přestože byl zmapován pouze zlomek celé sítě, výsledkem bylo přes všechna očekávání, zcela jiné rozdělení stupňů - mocninné. Přes 80% uzlů mělo méně než čtyři spojení, ale méně než 0.01% uzlů mělo více než tisíc spojení. Sítě, které se řídí mocninným rozdělením stupňů, nazvali Barabási a jeho kolegové bezškálovými sítěmi (scale-free network). Rozpoznání tohoto jevu vedlo k lepšímu porozumění šíření virů a epidemií nebo proč některé sítě fungují takřka beze změny i přes poruchu většiny jejich uzlů. Sociální a citační sítě se řadí do kategorie bezškálových sítí. Například autor vědecké literatury, jehož dílo je v dané oblasti známé, má velkou šanci, že bude citován dalšími autory, především těmi novými. Stejně tak osoba s velkým počtem přátel má velikou šanci, že bude představen novým lidem a rozšíří si tak svůj okruh přátel ještě více. Tomuto jevu v bezškálových sítí se říká „bohatší se stává bohatším“.

Rozvinutou disciplínou v oblasti sítí je analýza sociálních sítí, která se stala klíčovou technikou v moderní sociologii a stala se významnou v různých vědeckých oblastech (antropologie, biologie, komunikace, ekonomie, informační věda, geografie, historie, politologie, ...).

Analýza sociálních sítí zkoumá povahu vztahů (homofilie, multiplexita, vzájemnost, blízkost vztahů, ...), rozdělení vlastností v síti (míry centrality, hustota sítě, ...) nebo segmentaci (souvislost grafu, komponenty grafu, kliky, koeficient shlukování, ...).

3.1.1 Souvislost a komponenty grafu

Neorientovaný graf je souvislý, pokud pro každé jeho dva vrcholy u a v existuje alespoň jedna cesta z u do v. U neorientovaného grafu hovoříme o slabé souvislosti. Pro orientovaný uvažujeme silnou souvislost, protože přestože existuje cesta z u do v, není zaručeno, že existuje cesta z v do u. Slabě souvislý orientovaný graf znamená, že neorientovaný graf, který by vznikl nahrazením orientovaných hran neorientovanými (symetrizace grafu), by byl souvislý.

Komponenta maximálně souvislý podgraf. Jinak řečeno komponenta je

podgraf takový, že všechny jeho vrcholy jsou spojeny nějakou cestou. Komponentou je i samostatný vrchol.

Všechny slabě souvislé komponenty grafu najdeme pomocí jednoduchých algoritmů „prohledávání do šířky“ nebo „do hloubky“. Spuštění prohledávání najde celou komponentu, ve které se výchozí vrchol nachází. Pokud zaznamenáváme, které vrcholy byly nalezeny, a spustíme prohledávání ze všech nenalezených vrcholů, najdeme všechny komponenty.

Silně spojité komponenty nenajdeme pouhým prohledáním do šířky nebo do hloubky, ale použijeme sofistikovanější algoritmy (Kosarajův, Tarjanův, ...), které principově vycházejí z prohledávání do hloubky.

3.1.2 Klika v grafu

Klika (clique) grafu je úplný podgraf. To znamená, že všechny vrcholy kliky jsou spojeny přímo hranou.

V sociologii slovo klika popisuje skupinu dvou až dvanácti lidí, kteří jsou na sebe vázáni více než na jiné lidi v tomtéž prostředí (cite Neil Salkind - Encyclopedia of educational psychology). Klika je silněji spojená skupina lidí než sociální kruh.

Algoritmus pro nalezení největší kliky v grafu je přímočaré otestování n vrcholů podgrafu pro všech 2^L podgrafů grafu, kde L je horní limit velikosti podgrafu. Pokud je všech $\frac{n(n-1)}{2}$ párů vrcholů daného podgrafu spojených hranou, pak se jedná o kliku. Problém má exponenciální složitost, proto je horní hranice velikosti podgrafu ve výpočtech omezena na 20.

3.2 Významnost uzlů

Významnost autorů je jedním předmětem zájmu analýzy sociálních sítí. Kdybychom se měli rozhodnout, kterého člena sítě zvolit jako vůdce nebo přes které členy nejrychleji rozšíříme zprávu, koho bychom měli vybrat?

Velký díl k zodpovězení otázky relativní významnosti prvků definovali Freeman (1979), Bonacich (1972), jehož práce je spojena s Hubbellovo (1965) mírou sociometrického statusu, Coleman (1973) se svou mírou síly a Burt (1982) a jeho míra prestiže (cite Noah E. Friedkin, Theoretical Foundations for Centrality Measures). Významnost prvku bývá v sociální síti označována jako centralita a metody pro zjištění centrality jsou známé jako míry centrality (centrality measure). Původně byly vyvinuty v sociologickém kontextu pro analýzu sociálních sítí, ale jejich princip lze snadno zobecnit na obecný graf, proto můžeme využít těchto metod pro analýzu citačních nebo jiných komplexních sítí, které nemají čistě sociologický význam.

3.2.1 Ocenění významných autorů

Významní autoři vědecké literatury bývají za své dílo oceněni významnou cenou nebo zařazeni do seznamů významných členů.

Autory, kteří byli oceněni těmito cenami, můžeme považovat za významné a častokrát citované už jen proto, že přítomností jejich jména v seznamu oceněných prestižní cenou se dostanou do podvědomí mnoha jiných, zejména začínajících autorů.

ACM A.M. Turing Award

ACM A.M. Turing Award je ocenění ročně udělované skupinou ACM (Association for Computing Machinery) jedincům vybraným pro kontribuce technického ducha do výpočetního světa. [tur].

Turingova cena je brána jako nejvyšší vyznamenání v informatice a je lidově nazývána Nobelovou cenou pro informatiku [DPV08, p. 317].

ACM SIGMOD Edgar F. Codd Innovations Award

ACM SIGMOD Edgar F. Codd Innovations Award je ohodnocení životního díla skupinou ACM SIGMOD (Special Interest Group on Management of Data) za inovativní a vysoce ceněné kontribuce k rozvoji, porozumění a použití databázových systémů a databází [sig].

ACM Fellows

„The ACM Fellows Program“ byl založen v roce 1993, aby našel a ocenil vynikající členy ACM za jejich dílo v informatice a informační vědě a pro jejich významné kontribuce pro účel ACM. Členové ACM Fellows slouží jako význační kolegové, ke kterým ACM a jejich členové vzhlížejí jako k autoritám v době rozvoje informačních technologií [acm].

ISI Highly Cited highlighted

ISI Highly Cited je databáze často citovaných autorů v článcích posledního desetiletí, které byly publikovány institutem ISI (Institute for Scientific Information). Ten v dnešní době spadá pod agenturu Thomson Reuters, na jejíchž webových stránkách nalezneme seznam autorů ISI Highly Cited highlighted z let 2000 až 2008 napříč 21 vědeckými obory [hig].

3.3 Míry centrality

V sítích dopravní infrastruktury nás zajímá, po které cestě se nejrychleji a nejvýhodněji dostat z bodu A do bodu B. V sociálních a citačních sítích nemůžeme intuitivně hovořit o nějakých cestách mezi uzly, protože ani přesně nevíme, jak takovou cestu interpretovat. Nejkratší cesta mezi přáteli v sociální síti může znamenat přes které přátele se mezi nimi nejpravděpodobněji šíří informace. V sítích spolupráce vědeckých autorů se například setkáme s tzv. Erdősovým číslem, které vyjadřuje nejkratší vzdálenost mezi osobou a matematikem Paulem Erdősem v rámci spolupráce na vědeckých článcích v oboru matematiky.

Použijeme-li metody z dopravních sítí pro analýzu sociálních a citačních sítí, které v jádře spočívají v hledání nejkratších cest, setkáme se se dvěma nejznámějšími mírami centrality, a to closeness centrality a betweenness centrality.

Nechť cesta z bodu $u \in V$ do bodu $v \in V$ je střídající se posloupnost vrcholů a hran takových, že spojují předcházející a následující vrchol v této posloupnosti. Délka cesty je pak součet vah hran této cesty nebo pouze počet hran v případě neváženého grafu. Vzdálenost vrcholů $d_G(u, v)$ je délka nejkratší z cest, která spojuje vrcholy u a v.

Jiné míry jsou založeny na počtu spojení jednoho uzlu s ostatními uzly a nejkratší cesty neuvažují (degree, eigenvector).

3.3.1 Degree

Stupeň je počet hran spojených s uzlem. Pro orientovaný graf můžeme uvažovat vstupní (indegree) a výstupní stupeň (outdegree) vrcholu nebo obecný stupeň (degree), tedy součet těchto dvou. Vstupní stupeň se často označuje jako deg^- a výstupní jako deg^+. Nechť C_D označuje míru centrality degree a C_{Din}, C_{Dout} centralitry indegree a outdegree, respektive. Pak můžeme vyjádřit hodnoty centrality pomocí matice sousednosti $\mathbf{A}$.

$$C_{Din}(u) = deg^-(u) = \sum_{v \in V} \mathbf{A}_{uv} \tag{3.1}$$

$$C_{Dout}(u) = deg^+(u) = \sum_{v \in V} \mathbf{A}_{vu} \tag{3.2}$$

$$C_D(v) = C_{Din} + C_{Dout} \tag{3.3}$$

Pokud uvažujeme pouze vstupní stupeň, vypočtená hodnota určuje významnost uzlu, kdežto výstupní stupeň ukazuje jakousi společenskost či otevřenost uzlu.

Degree centrality je výpočetně velmi jednoduchý způsob, jak změřit významnost prvku v síti. Tato metoda je však příliš jednoduchá, protože do výpočtu hodnoty centrality nezahrnuje uzly, které jsou od daného uzlu vzdálenější než jeden krok. Tento fakt je známý problém a důvod pro zavedení dalších a složitějších metod pro výpočet významnosti.

3.3.2 Eigenvector centrality

Eigenvector centrality (také známá jako Gould's index of accessibility of a Network (Linear Algebra with Applications: Alternate Edition by Gareth Williams) nebo Bonacich's centrality (Robert A. Hanneman, Mark Riddle - Introduction to social network methods)), je míra vlivu vrcholu v grafu, která doslova znamená „Důležitý uzel má důležité sousedy" (cite An introduction to Centrality measures, Zweig, Iyengar, 2010). Hodnotu vlivu získáme z vlastního vektoru x matice sousednosti grafu:

$$\mathbf{Ax} = \lambda\mathbf{x} \tag{3.4}$$

A je matice sousednosti, **x** je vlastní vektor matice **A** a řešením této rovnice. Rovnice má více řešení. Ke každému řešení náleží vlastní číslo λ. Pro měření významnosti nás však zajímá pouze to řešení, které má pouze nezáporné hodnoty. Podle Perron-Frobeniovy věty pro každou nezápornou primitivní matici existuje právě jedno takové řešení, které zároveň patří k největšímu vlastnímu číslu λ [LM06].

Rovnici můžeme rozepsat z maticového tvaru do jednotlivých složek:

$$x_u = \frac{1}{\lambda}\sum_{v\in G}\mathbf{A}_{uv}x_v \tag{3.5}$$

Kde x_u je prvek vlastního vektoru **x** náležící vrcholu u a $\mathbf{A}_{uv}$ je prvek matice sousednosti **A**, který leží na řádku u a sloupci v.

$$x_{ui+1} = \frac{1}{\lambda}\sum_{v\in G}\mathbf{A}_{uv}x_{vi} \tag{3.6}$$

V tomto rekurentním tvaru je vidět předpis pro iterační výpočet eigenvector centrality. Algoritmus se nazývá mocninná metoda, která se používá pro řešení problému vlastních čísel v numerické matematice. Výsledkem mocinné metody je dominantní vlastní číslo a odpovídající vlastní vektor. Pro eigenvector centrality nás zajímá právě tohle řešení a žádné jiné.

Z druhé rovnice si navíc povšimneme, že se jedná o přímé rozšíření degree centrality (3.1). Výsledek předchozí iterace použijeme jako vstup do následující a iterujeme tak dlouho, dokud nedosáhneme požadované přesnosti.

PageRank

V roce 1998 vyvinuli Sergey Brin a Larry Page algoritmus PageRank (nesoucí jméno druhého autora) jako součást výzkumu na novém druhu webového vyhledávače (cite něco). PageRank přiřazuje relativní hodnocení webovým stránkám podle hypertextových odkazů z jiných webových stránek, které na ně směřují, a podle jejich PageRankové významnosti. Sama definice je rekurzivní a po nahlédnutí na vzorec zjistíme, že se jedná o rozšířenou variantu algoritmu pro eigenvector centrality.

$$x_{ui+1} = \frac{1-d}{|V|} + d\sum_{v \in V} \mathbf{A}_{uv} \frac{x_{vi}}{deg^+(v)} \tag{3.7}$$

A je opět matice sousednosti, V je množina vrcholů a $deg^+(v)$ je výstupní stupeň vrcholu v. V původní myšlence, kdy PageRank modeluje chování náhodného surfaře, damping factor d je pravděpodobnost, že daný surfař přestane opakovaně klikat na odkazy, které najde na webové stránce, na kterou se dostal z předchozího odkazu, a otevře zcela novou stránku, ze které povede podobný sled surfování přes odkazy. Damping factor bývá ze zkušenosti nastavován na 85%.

Hodnota PageRanku je z matematického hlediska pravděpodobnost, že surfař, který náhodně kliká na odkazy, se dostane na konkrétní stránku. Součet hodnot PageRanku všech uzlů v síti je tedy 1, protože PageRank je rozdělení pravděpodobnosti.

Jedním problémem algoritmu PageRank jsou „visící uzly“ (dangling nodes), tj. uzly bez výstupních hran. Protože musíme v každé iteraci algoritmu zachovat vlastnost rozdělení pravděpodobnosti, tj. suma všech pravděpodobností je 1, je třeba zajistit, aby se přenášená hodnota mezi iteracemi neztrácela právě v uzlech bez výstupních hran. Problém lze řešit tak, že se tyto uzly z výpočtu vynechají, nebo přidáním zpětných odkazů z těchto uzlů zpět do sítě. V každé iteraci předem vypočítáme součet hodnot PageRanku, které by se ztratily v uzlech bez výstupních hran (D). Tahle hodnota je v rámci iterace konstantní a pouze ji rovnoměrně rozdělíme mezi uzly sítě (s váhou $1/|V|$).

$$x_{ui+1} = \frac{1-d}{|V|} + d\sum_{v \in V} \mathbf{A}_{uv} \frac{x_{vi}}{deg^+(v)} + \frac{1}{|V|}\sum_{w \in D} x_{wi} \tag{3.8}$$

Algorithm 1 PageRank

$x_0[v] \leftarrow \frac{1}{|V|}$, $v \in V$ ▷ Uzly začínají se stejnou pravděpodobností
for $i \leftarrow 0, K$ **do** ▷ Iterujeme K-krát, dokud není dosažena požadovaná přesnost
    $s \leftarrow 0$ ▷ s je suma PageRanků pro dangling nodes
    for $v \in V$ **do**
        if $deg^+(v) = 0$ **then**
            $s \leftarrow s + x_i[v]$
        end if
    end for
    $x_{i+1}[v] \leftarrow \frac{s}{|V|}$, $v \in V$ ▷ Každý uzel začíná s přebytkem z dangling nodes
    for $v \in V$ **do**
        for $k \in inLinks[v]$ **do**
            $x_{i+1}[v] \leftarrow x_{i+1} + \frac{x_i[k]}{deg^+(k)}$
        end for
    end for
    $x_{i+1}[v] \leftarrow \frac{1-d}{|V|} + dx_{i+1}[v]$, $v \in V$ ▷ Přidání damping factoru
end for

Přestože je PageRank původně určený pro webovou síť, lze ho použít na jakoukoliv orientovanou váženou i neváženou síť, tedy i na sociální a citační sítě, o kterých je zde řeč. Pro neorientovaný graf je hodnota PageRanku pro jednotlivé uzly velmi blízká stupňům grafu, ale ne totožná (cite icola Perra and Santo Fortunato.; Fortunato (September 2008). "Spectral centrality measures in complex networks")

3.3.3 Closeness centrality

Closeness neboli „blízkost“ je definována jako převrácená hodnota míry farness, tedy „dalekost“. Dalekost je součet všech vzdáleností od uzlu do všech ostatních, tzn. $f(u) = \sum_{v \in V} d_G(u, v)$ a $c(u) = \sum_{v \in V} \frac{1}{d_G(u,v)}$. Podle jiné definice je closeness převrácená hodnota průměrné nejkratší cesty. V podstatě se od předchozí příliš neliší, protože průměrná nejkratší cesta je rovna $\frac{1}{|V|-1} \sum_{v \in V} d_G(u, v)$ a closeness podle této definice:

$$c(u) = \frac{|V| - 1}{\sum_{v \in V} d_G(u, v)}$$

Pro obě definice platí, že čím vyšší hodnota $c(u)$, tím je uzel u významnější podle této míry. Zde se budeme soustředit na druhou definici, protože je častou volbou autorů zabývajících se touto problematikou a existuje pro ni aproximační algoritmus, který si zde uvedeme.

Closeness, stejně jako ostatní míry centrality, modeluje rozptýlení informace napříč sítí. Výše uvedené klasické definici je vytýkáno, že pro přenos informace uvažuje pouze nejkratší cesty, které nejsou vždy jedinou komunikační cestou v síti. Alternativu navrhli Noh a Rieger (2004), kde namísto nejkratších cest používají náhodné procházky (random walk closeness centrality). Náhodná procházka (random walk) je cesta, kde v každém uzlu je náhodně vybrán směr dalšího postupu.

Příkladem může být oběh mincí mezi lidmi. Tento jev nemá s nejkratšími cestami mnoho společného, proto je vhodnější ho modelovat náhodnými procházkami. Oproti tomu například poštovní zásilky očividně cestují po nejkratších cestách. Pokud uvažujeme citační sítě, nemáme jasnou představu o významu náhodných procházek nebo nejkratších cest jako v případě mince nebo dopisu. I přesto očekáváme vysokou podobnost této metody s ostatními.

Nevýhodou closeness centrality je nutnost uvažovat souvislý graf, tedy takový, který obsahuje pouze jednu komponentu. Pokud by měl více komponent, pak by vždy existovala cesta s nekonečnou vzdáleností. Hodnota farness by pak byla automaticky nekonečná a closeness, tedy převrácená hodnota, by byla nulová.

Existuje několik upravených definic, které se mají vypořádat s problémem konektivity a druhotně jsou numericky stabilnější. Jedna z nich zaměňuje převrácenou hodnotu součtu vzdáleností za součet převrácených hodnot vzdáleností $c(u) = \sum_{v \in V} \frac{1}{d_G(u,v)}$ (Opsahl) a druhá $c(u) = \sum_{v \in V} 2^{-d_G(u,v)}$ (Dangalchev). Přesto se nejvíce používá původní definice closeness a výpočet se omezí na největší komponentu.

Algoritmus

Closeness pro všechny vrcholy můžeme přesně vypočítat v čase $O(|V||E| + |V|^2 \log |V|)$, kde V a E jsou množiny vrcholů a hran sítě (cite JO77, FT87).

Algoritmus vychází z definice, tedy vyřeší problém všech párů nejkratších cest, čímž rovnou získá hodnoty farness $f(u) = \sum_{v \in V} d_G(u, v)$ a zjištění closeness je poté triviální podle jedné z výše uvedených definic. Výše uvedená složitost platí při použití Dijkstrova algoritmu pro všechny páry cest.

Algorithm 2 Closeness

```
1: for s ∈ V do
2:     f ← 0                                              ▷ Farness
3:     d[v] ← ∞, v ∈ V                  ▷ Zpočátku jsou uzly nedosažitelné
4:     d[u] ← 0
5:     Q ← {s}          ▷ Prioritní fronta Q začíná se zdrojovým vrcholem
6:     while Q ≠ ∅ do                          ▷ Dokud není fronta prázdná
7:         u ← extract-min(Q)        ▷ Vytáhneme uzel s min. vzdáleností d
8:         f ← f + d[u]                      ▷ Zvýšíme farness pro nový uzel
9:         for sousedící vrchol v ∈ Adj[u] do      ▷ Přidáme nově nalezené
   vrcholy
10:            if d[v] > d[u] + w(u, v) then
11:                d[v] ← d[u] + w(u, v)
12:                Q ← Q ∪ {v}
13:            end if
14:        end for
15:    end while
16:    c[s] ← (n−1)/f                ▷ Closeness je převrácená hodnota farness
17: end for
```

Výpočet closeness lze snadno paralelizovat, jelikož výpočet stromu nejkratších cest je nezávislá úloha pro každý z $|V|$ vrcholů.

Pro rozsáhlé sítě s miliony uzlů (sociální sítě k dnešnímu datu) je tato metoda příliš náročná. Eppstein a Wang vyvinuli aproximační algoritmus s náročností $O(\frac{\log|V|}{\epsilon}^2(|V|\log|V|+|E|))$ s chybou $\epsilon\delta$ pro převrácenou hodnotu closeness (s pravděpodobností alespoň $1-\frac{1}{|V|}$), kde $\epsilon > 0$ a δ je poloměr sítě (nejdelší z nejkratších cest). Na základě tohoto aproximačního algoritmu byl vytvořen jiný aproximační algoritmus pro nalezení k nejvýznamnějších uzlů hodnocených podle closeness centrality TOPRANK (cite Okamoto, Chen, Li 2008 Ranking of Closeness Centrality for Large-Scale Social Networks).

Aproximace

Algoritmus TOPRANK (Okamoto, Chen, Li) najde prvních k nejvýznamnějších uzlů s vysokou přesností a pro každý z nich přesnou hodnotu closeness. Algoritmus pracuje s myšlenkou, že zjistíme přibližné pořadí uzlů tak, že pro jeden strom nejkratších cest nebudeme počítat se všemi koncovými uzly, ale jen s dostatečně velkým vzorkem této množiny. Přesné hodnoty closeness dosáhneme použitím exaktního algoritmu, který použijeme jen na nejvýznam-

nější uzly získané z prvního aproximovaného kroku. Klíčovou otázkou je, kolik nejvýznamnějších uzlů musíme uvažovat, aby se jednalo o dostatečně přesný výsledek. Autoři algoritmu uvádějí tento algoritmus s heuristikou, která najde přibližně místo, ve kterém je vhodné výpočet ukončit a považovat za dostatečně přesný. Sami uvádějí, že tento algoritmus je pouze první krok k návrhu efektivnějšího způsobu jak najít prvních k nejvýznamnějších uzlů (cite to co nahoře).

3.3.4 Betweenness centrality

Betweenness je druhá metoda, která modeluje šíření informace sítí pomocí nejkratších cest. Princip betweenness spočívá ve zvýhodnění uzlů, přes kterou teče nejvíce informace. Pokud uzel A komunikuje s uzlem C, můžeme tvrdit, že uzel B, který leží mezi nimi, bude mít roli prostředníka. Být tímto prostředníkem mezi více uzly intuitivně napovídá, že takový uzel bude centrální. „Čím více lidí na mně závisí k vytvoření spojení s jinými lidmi, tím mám větší moc“ (cite introductino to social network methods). Betweenness měří, na kolika nejkratších cestách se uzel nachází. Více se ale setkáme s definicí, kde do sumy zahrneme poměr cest, na kterých se uzel nachází, k celkovému počtu cest mezi dvěma uzly (Freeman, 1977; Anthonisse, 1971, Brandes):

$$C_(v) = \sum_{s \in V} \sum_{t \in V \setminus s} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$C_b(v)$ značí hodnotu betweenness centrality pro uzel v, V množinu všech uzlů grafu, σ_{st} je počet nejkratších cest mezi uzly s a t a $\sigma_{st}(v)$ je počet nejkratších cest, které navíc procházejí uzlem v.

Normalizovaný betweenness je hodnota v intervalu od 0 do 1, kterou získáme tak, že betweenness vydělíme celkovým počtem možných cest, tj. $(|V|-1)(|V|-2)$ pro orientované grafy a $\frac{(|V|-1)(|V|-2)}{2}$ pro neorientované grafy. Normalizované hodnoty metod centralit jsou nezávislé na velikosti grafu (cite Douglas R. WHite, Stephen P. Borgatti - Betweenness centrality measures for directed graphs)

$$C_b(v) = \frac{1}{(|V|-1)(|V|-2)} \sum_{s \in V} \sum_{t \in V \setminus s} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Vznik betweenness je připisován sociologovi Lintonu Freemanovi (Freeman 77) a byl původně definován pro neorientované grafy.

Brandesův algoritmus

Ve své práci Ulrik Brandes zmiňuje (do té doby nejrychlejší) algoritmus pro výpočet betweenness centrality s časovou náročností $\theta(|V|^3)$ a $\theta(|V|^2)$ paměťovými nároky. Tento způsob přistupuje k problému nekratších cest způsobem all-pair shortest paths. Brandesův způsob využívá algoritmu pro nalezení nejkratších cest z jednoho bodu, kde výsledný algoritmus pracuje s paměťovou náročností $O(|V| + |E|)$ a běží v čase $O(|V||E|)$ pro nevážený graf nebo $O(|V||E| + |V|^2 \log |V|)$ pro vážený graf.

Brandes ve své práci o algoritmu uvádí pseudokód pro nevážený graf, který je následně snadné pozměnit pro vážený graf drobnými úpravami a zaměněním obyčejné fronty za prioritní frontu; kompletní důkaz správnosti algoritmu a porovnání standardního algoritmu s tímto (TODO cite Brandes).

Algorithm 3 Brandesův algoritmus

1: $b[v] \leftarrow 0$, $v \in V$
2: **for** $s \in V$ **do**
3: $S \leftarrow$prázdný zásobník
4: $P[w] \leftarrow$prázdný seznam, $w \in V$
5: $\sigma[t] \leftarrow 0$, $t \in V$
6: $\sigma[s] \leftarrow 1$
7: $d[t] \leftarrow -1$, $t \in V$
8: $d[s] \leftarrow 0$
9: $Q \leftarrow \{s\}$
10: **while** $Q \neq \varnothing$ **do**
11: $v \leftarrow dequeue(Q)$
12: $push(S, v)$
13: **for** sousedící vrchol $w \in Adj[v]$ **do**
14: **if** $d[w] < 0$ **then**
15: $enqueue(Q, w)$
16: $d[w] \leftarrow d[v] + 1$
17: **end if**
18: **if** $d[w] = d[v] + 1$ **then** ▷ Nejkratší cesta do w přes v?
19: $\sigma[w] \leftarrow \sigma[w] + \sigma[v]$
20: $push(P[w], v)$
21: **end if**
22: **end for**
23: **end while**
24: $\delta[v] \leftarrow 0$,$v \in V$ ▷ δ je závislost uzlu s na ostatních
25: **while** $S \neq \varnothing$ **do** ▷ S vrátí vrcholy v pořadí s nezvyšující se vzdáleností od s
26: $w \leftarrow pop(S)$
27: **for** $v \in P[w]$ **do**
28: $\delta[v] \leftarrow \delta[v] + \frac{\sigma[v]}{\sigma[w]}(1 + \delta[w])$
29: **end for**
30: **if** $w \neq s$ **then** ▷ Betweenness je součet dependencí $\delta[w]$
31: $b[w] \leftarrow b[w] + \delta[w]$
32: **end if**
33: **end while**
34: **end for**

Algoritmus lze paralelizovat stejně jako algoritmus pro closeness centrality, pokud zajistíme synchronizaci vláken při přístupu k hodnotám betweenness.

Narozdíl od algoritmu pro closeness, kde výpočet pro jeden uzel ovlivňuje hodnotu closeness pouze pro tento uzel, u betweenness výpočet vycházející z jednoho uzlu ovlivňuje hodnotu betweenness potencionálně i pro všechny ostatní uzly.

Aproximace

I přes použití rychlejšího Brandesova algoritmu je výpočet betweenness centrality příliš náročný výpočet pro sítě reálného světa (např. biologické, dopravní nebo webové sítě) a pokud nám jde více o relativní pořadí uzlů podle hodnoty betweenness než o hodnotu samotnou, lze oželit přesný výpočet přibližným, který příliš nezmění výsledné umístění v žebříčku nejvýznamnějších uzlů.

Bader, Kintali, Madduri, Mihail ukazují aproximační algoritmus pro betweenness s odhadem chyby. Myšlenkou je jednoduchá lineární extrapolace Brandesova algoritmu, pokud do výpočtu zahrneme pouze náhodný vzorek namísto celé množiny vrcholů. Nechť k je velikost vzorku množiny vrcholů, se kterým počítáme, pak extrapolovaná hodnota betweenness je $\frac{|V|S}{k}$, kde S je vypočtená přibližná hodnota.

3.4 Hledání nejkratších cest

Hledání nejkratších cest v grafu je historicky starý problém, jehož matematický výzkum přišel relativně pozdě v porovnání s jinými problémy kombinatorické optimalizace (nejmenší kostra grafu, přiřazovací a dopravní problém). Pravděpodobně byl výzkum opožděn, protože se jedná o intuitivní a relativně jednoduchý problém, ale jakmile se dostal do středu zájmu, bylo nezávisle na sobě nalezeno několik metod řešení různými lidmi (Shimbel [1955], Ford [1956], Dantzig [1958], Bellman [1958], Moore[1959], Dijkstra [1959]).(cite schrijver alexander ON THE HISTORY OF THE SHORTEST PATH PROBLEM)

Z hlediska metod řešení můžeme uvažovat několik kategorií algoritmů - nalezení všech párů nejkratších cest (all-pairs shortest paths problem), nalezení nejkratší cesty mezi počátečním a koncovým vrcholem (source-target) nebo nalezení stromu nejkratších cest, máme-li zadán počáteční vrchol (single source shortest path problem).

3.4.1 Single source shortest path

Pokud hledáme pouze jednu cestu mezi dvěma vrcholy (source-target), nemusíme počítat celý strom nejkratších cest, ale můžeme zastavit výpočet při dosažení požadovaného vrcholu.

BFS

Prohledávání do šířky z jednoho bodu (breadth first search) je algoritmus, který najde nejkratší cesty z jednoho bodu do všech ostatních v případě neváženého grafu v čase $O(E)$. Pro vážený graf by zjistil nejkratší cesty, kde metrika vzálenosti by byla počet skoků mezi uzly.

Algorithm 4 Prohledávání do šířky

```
1: function BFS(G, s)
2:     d[v] ← ∞,v ∈ V                    ▷ Uzly jsou zpočátku nedosažitelné
3:     d[s] ← 0
4:     p[v] ← NIL,v ∈ V                  ▷ A nemají implicitně žádné předchůdce
5:     Q ← {s}                           ▷ Fronta začíná s počátečním vrcholem
6:     while Q ≠ ∅ do                    ▷ Dokud není prázdná
7:         u ← dequeue(Q)
8:         for sousedící vrchol v ∈ Adj[u] do
9:             if d[v] = −1 then         ▷ Pro všechny nově objevené vrcholy
10:                d[v] ← d[u] + 1       ▷ Nově vypočtená vzdálensot
11:                Q ← Q ∪ {v}           ▷ Přidáme nově objevené uzly do fronty
12:                p[v] ← u              ▷ Předchůdce uzlu v je u
13:            end if
14:        end for
15:    end while
16:    return p
17: end function
```

Bellman-Fordův algoritmus

Bellman-Fordův algoritmus je aplikací dynamického programování na nejkratší cesty z jednoho vrcholu do všech ostatních. Jeho využití najdeme zejména tam, kde se mohou objevit záporné váhy hran. Časová náročnost je $O(|V||E|)$.

Dijkstrův algoritmus

Dijkstrův algoritmus řeší „single source shortest path problém“ pomocí greedy (žravé) strategie. Časová náročnost běžné implementace $O((|E| + |V|) \log |V|)$ je lepší než u Bellman-Fordova algoritmu, ale Dijkstrova greedy strategie funguje, pouze pokud v grafu neexistují záporně ohodnocené hrany. Časová náročnost závisí především na implementaci klíčové datové struktury prioritní

fronty, která v každém kroku vybírá uzel s nejmenší vzdáleností od zdrojového vrcholu (greedy strategie). V případě použití obyčejného spojového seznamu jako prioritní fronty získáme kvadratickou náročnost $O(|V|^2)$. Nejčastěji se setkáme s prioritní frontou implementovanou pomocí binární haldy, která zajišťuje čas běhu právě $O((|E|+|V|)\log|V|)$. Pomocí Fibonacciho haldy získáme doposud nejrychlejší Dijkstrův algoritmus s časem běhu $O(|E|+|V|\log|V|)$, ale pro běžné účely (grafy menší než miliony vrcholů) je nejvýhodnější binární halda (cite CLRS, Fredman & Tarjan). Pro nevážený graf je Dijkstrův algoritmus ekvivalentní prohledávání do šířky.

Dijkstrův algoritmus je klíčovým pro všechny implementované metody založené na nejkratších cestách, proto uvedeme pseudokód (cite CLRS):

Algorithm 5 Dijkstrův algoritmus

```
1: function DIJKSTRA(G, w, s) ▷ s je zdrojový vrchol a w je váhová funkce
2:     d[v] ← ∞,v ∈ V                    ▷ Vrcholy jsou zpočátku nedosažitelné
3:     d[s] ← 0
4:     p[v] ← NIL,v ∈ V                  ▷ Předchůdci vrcholů zpočátku neexistují
5:     Q ← V                             ▷ Prioritní fronta Q obsahuje všechny vrcholy
6:     while Q ≠ ∅ do                    ▷ Postupujeme, dokud není prázdná
7:         u ← extract-min(Q)            ▷ Vytáhneme z fronty vrchol s nejmenší
   hodnotou d[u]
8:         for sousedící vrchol v ∈ Adj[u] do
9:             if d[v] > d[u] + w(u, v) then
10:                d[v] ← d[u] + w(u, v)          ▷ Relaxace hrany {u, v}
11:                p[v] ← u
12:            end if
13:        end for
14:    end while
15:    return p                                   ▷ Strom předchůdců
16: end function
```

Výsledkem je strom předchůdců, který reprezentuje strom nejkratších cest. Princip Dijkstrova algoritmu se objevuje i u algoritmů pro closeness, betweenness a BFS. V tomto případě do prioritní fronty prvotně zahrneme všechny vrcholy. Jinou možností je přidávat je postupně, aby vybírání z prioritní fronty (*extract-min*(Q)) bylo rychlejší. Tento způsob je použit v algoritmu pro closeness.

3.4.2 All-pair shortest paths

Do této kategorie spadají maticové metody, tj. graf je zadán jako matice sousednosti nebo matice sousednost s váhami hran.

Shimbelova metoda

Shimbelova metoda (1955) používá upravené maticové násobení k získání $|V|$-té mocniny matice sousednosti. Celková časová náročnost je $O(|V|^4)$, protože provedeme $|V|$ „násobení“ čtvercové matice o složitosti $O(|V|^3)$. Shimbelovo upravené násobení nahrazuje sčítání a násobení za minimum a sčítání:

$$x + y \equiv \min(x, y)$$
$$xy \equiv x + y$$

Floyd-Warshallův algoritmus

Floyd-Warshallův algoritmus snižuje časovou náročnost na $O(|V|^3)$ použitím dynamického programování. Graf je opět zadán jako vážená matice sousednosti. Rekurentní vzorec dynamického programování pro tento algoritmus je:

$$d_0(u, v) = \mathbf{A}_{uv}$$
$$d_{k+1}(u, v) = \min(d_k(u, v), d_k(u, k) + d_k(k, v))$$

Jednodušše zkoušíme, zda je kratší cesta mezi vrcholy u a v, kterou již známe, nebo jiná cesta za použití nějakého vrcholu k, který leží mezi nimi. Výpočet provádíme pro všechny páry vrcholů pro každý vrchol k ($|V|^2|V|$).

Johnsonův algoritmus

Johnsonův algoritmus nepatří mezi maticové metody, protože využívá metod single source shortest path pro všechny vrcholy. V principu jednodušše použijeme Dijkstrův algoritmus pro každý vrchol zvlášť, ale dovolujeme váhy hran i záporné. V případě záporných hran je nutné provést transformaci vah pomocí Bellman-Fordova algoritmu, která v grafu nepozmění nejkratší cesty. (cite CLRS)

3.5 Ostatní míry významnosti autorů

Míry centrality jsou určeny pro relativní seřazení uzlů pro obecný graf. Jelikož zde pracujeme speciálně s citačními sítěmi, zajímá nás, jestli existují i jiné metody ohodnocení autorů, které ani nevyžadují konstrukci citačních sítí, ale vyžadují pouze údaje v databázi.

Setkáme se s metodami AWCR, AW-index, Eigenfactor, Egghův g-index, E-index, I10-index, R-impact, Wu Index a zejména H-index (cite http://hlwiki.slais.ubc.ca/index.php/ který zde byl implementován jako jediná metoda, která nespadá pod míry centrality.

3.5.1 H-index

H-index je metoda pro měření produktivity autora a významu jeho publikované vědecké práce. Metoda byla navržena fyzikem Jorge E. Hirschem pro zjišťování relativní významnosti vědců publikujících v oboru teoretické fyziky. Původní Hirschova definice zní:

> *Vědec má index h pokud h z jeho N_p publikací má každá alespoň h citací a žádná ze zbylých $(N_p - h)$ publikací nemá více než h citací.*

Tato metoda je specifická pro citační sítě a nelze ji aplikovat na sociální nebo obecnou komplexní síť.

Výpočet probíhá tak, že seřadíme autorovy publikace P sestupně podle počtu citací $c : P \mapsto \mathbb{N}$ a poté od začátku tohoto seřazeného seznamu hledáme tu publikaci, jejíž pořadové číslo v tomto seznamu je nižší nebo rovné počtu jejích citací. Toto číslo je pak h-index autora.

Algorithm 6 H-index

```
1: function H-INDEX(P, c)
2:     sort(P, key ← c)    ▷ Seřadíme publikace sestupně podle počtu citací
3:     h ← 1
4:     for p ∈ P do
5:         if c(p) ≤ h then    ▷ Bod, kde počet citací vyrovná h, je h-index autora
6:             break
7:         end if
8:         h ← h + 1
9:     end for
10:    return h
11: end function
```

4 Výsledky

Pro zjištění, jestli se výsledky implementovaných metod shodují s uvedenými oceněními, použijeme metodu součtu pořadí oceněných autorů, viz tabulka 4.6. Tzn. že pro jedno ocenění sečteme pořadí všech autorů, kteří byli oceněni danou cenou. Tuto jednoduchou míru můžeme porovnat pouze mezi jednotlivými metodami pro jedno ocenění, ale ne mezi různými oceněními pro jednu metodu. Jednoduše protože např. Turingova cena je ve výsledcích udělena pouze několika autorům, z čehož plyne malý součet pořadí, kdežto velké množství autorů je členy ACM Fellows, tím pádem velký součet pořadí oceněných.

Dále nás zajímá, zda-li jsou metody mezi sebou podobné, či nikoli. Pro porovnání metod mezi sebou je použit Spearmanův koeficient pořadové korelace.

4.1 Spearmanův koeficient pořadové korelace

Spearmanův koeficient je klasický Pearsonův koeficient korelace, který je aplikovaný na proměnné s pořadím (cite Myers, Jerome L.; Well, Arnold D. (2003), Research Design and Statistical Analysis (2nd ed.), Lawrence Erlbaum, pp. 508).

Koeficient korelace obecně dosahuje hodnot od -1 do 1, přičemž hodnota 1 znamená naprostou lineární závislost mezi porovnávanými proměnnými tak, že s rostoucí první proměnnou roste i druhá proměnná. Hodnota -1 znamená rovněž naprostou lineární závislost, ale při rostoucí jedné proměnné druhá proměnná klesá. Hodnota 0 znamená kompletní náhodnost či nezávislost mezi měřenými proměnnými.

4.2 Porovnání implementovaných metod

Tabulka 4.2 ukazuje vypočtené Spearmanovy koeficienty pořadové korelace mezi všemi páry implementovaných metod. Do výpočtu koeficientu korelace byly zahrnuty pouze ty uzly, které mají alespoň jednu hranu. Pokud by byly zachovány, většina koeficientů korelace by byla velmi vysoká (větší než 0.95), protože většina vrcholů, zejména v databázi DBLP, jsou osamocené vrcholy. Protože jsou výsledky seřazeny podle hodnoty centrality a poté podle jména autora, všechny tyto izolované vrcholy, které nijak nepřispívají k hodnotě centrality, jsou pro všechny metody umístěné na posledních pozicích v tomtéž pořadí. Důsledkem je vysoký koeficient korelace, který nám však nic nevypoví o podobnosti výsledků metod.

Jelikož výpočet pro centralitu closeness ve všech implementovaných variantách (vstupní hrany, výstupní hrany, vážený closeness) byl proveden pouze na největší silně spojité komponentě, nelze přímo provést srovnání pomocí koeficientu korelace s výsledky metod, které byly aplikovány na celý graf, protože se jedná o dvě neporovnatelné proměnné. Proto byli z výsledků metod pro celý graf odebráni ti autoři, kteří se nenacházejí v hlavní silně spojité komponentě. Pro tyto žebříčky bylo provedeno stejné srovnání pomocí Spearmanova koeficientu korelace pořadí křížově mezi všemi metodami, viz tabulka 4.3

Tabulka 4.1 popisuje zkratky metod, časy běhu pro sítě DBLP a CiteSeer a velikost největší kliky v symetrizovaném grafu autorů. Největší klika je nalezena z top 20 autorů pro každou metodu. Výpočty byly vykonány na Intel®Core™2 Quad Q8400 přetaktovaném na 3.1 GHz s 4 GB fyzické paměti. Betweenness centrality bylo vypočteno paralalně na čtyřech jádrech, zbylé metody byly relativně časově nenáročné, proto nebylo nutné paralelizovat výpočet.

zkratka	metoda	t_{DBLP}	$t_{CiteSeer}$	klika$_{DBLP}$[1]	klika$_{CiteSeer}$[2]
hi	H-index	0:00:04:627		17	
ideg	indegree	0:00:00:049		19	
odeg	outdegree	0:00:00:036		16	
deg	degree	0:00:00:036		19	
wideg	vážený indegree	0:00:00:099		18	
wodeg	vážený outdegree	0:00:00:046		17	
wdeg	vážený degree	0:00:00:062		17	
pr	PageRank	0:00:00:376		18	
btw	betweenness	0:24:56:814		19	
btwA	aproximace betweenness	0:06:51:216		19	
wBtwA	aproximace váženého betweenness	0:08:54:073		16	
ic	closeness pro vstupní hrany	0:00:05:583		19	
oc	closeness pro výstupní hrany	0:00:11:313		16	
wic	vážený closeness pro vstupní hrany	0:01:14:054		18	

Tabulka 4.1: Implementované metody

[2]Největší klika mezi prvními 20 autory

	hi	ideg	odeg	deg	wideg	wodeg	wdeg	pr	btw	btwA	wBtwA
hi	-	0.571	0.252	0.508	0.596	0.258	0.525	0.533	0.517	0.513	0.520
ideg	0.571	-	0.173	0.669	0.988	0.178	0.669	0.909	0.880	0.873	0.830
odeg	0.252	0.173	-	0.670	0.186	0.998	0.662	0.102	0.121	0.131	0.134
deg	0.508	0.669	0.670	-	0.673	0.670	0.992	0.570	0.566	0.573	0.529
wideg	0.596	0.988	0.186	0.673	-	0.192	0.687	0.898	0.877	0.870	0.849
wodeg	0.258	0.178	0.998	0.670	0.192	-	0.665	0.108	0.126	0.136	0.140
wdeg	0.525	0.669	0.662	0.992	0.687	0.665	-	0.570	0.573	0.579	0.549
pr	0.533	0.909	0.102	0.570	0.898	0.108	0.570	-	0.818	0.810	0.754
btw	0.517	0.880	0.121	0.566	0.877	0.126	0.573	0.818	-	0.993	0.929
btwA	0.513	0.873	0.131	0.573	0.870	0.136	0.579	0.810	0.993	-	0.924
wBtwA	0.520	0.830	0.134	0.529	0.849	0.140	0.549	0.754	0.929	0.924	-

Tabulka 4.2: Tabulka korelací pro neizolované uzly

	hi	ideg	odeg	deg	wideg	wodeg	wdeg	pr	btw	btwA	wBtwA	ic	oc	wic
hi	-	0.718	0.551	0.698	0.752	0.579	0.721	0.674	0.657	0.656	0.679	0.626	0.486	0.674
ideg	0.718	-	0.509	0.813	0.984	0.527	0.804	0.940	0.918	0.917	0.865	0.890	0.467	0.844
odeg	0.551	0.509	-	0.877	0.532	0.980	0.865	0.389	0.458	0.457	0.505	0.435	0.892	0.493
deg	0.698	0.813	0.877	-	0.821	0.873	0.984	0.708	0.747	0.747	0.753	0.720	0.783	0.734
wideg	0.752	0.984	0.532	0.821	-	0.557	0.831	0.922	0.905	0.904	0.889	0.871	0.487	0.887
wodeg	0.579	0.527	0.980	0.873	0.557	-	0.889	0.410	0.475	0.475	0.529	0.453	0.884	0.527
wdeg	0.721	0.804	0.865	0.984	0.831	0.889	-	0.701	0.740	0.739	0.766	0.712	0.780	0.763
pr	0.674	0.940	0.389	0.708	0.922	0.410	0.701	-	0.894	0.894	0.815	0.891	0.340	0.803
btw	0.657	0.918	0.458	0.747	0.905	0.475	0.740	0.894	-	0.998	0.918	0.961	0.422	0.866
btwA	0.656	0.917	0.457	0.747	0.904	0.475	0.739	0.894	0.998	-	0.917	0.960	0.421	0.864
wBtwA	0.679	0.865	0.505	0.753	0.889	0.529	0.766	0.815	0.918	0.917	-	0.876	0.463	0.930
ic	0.626	0.890	0.435	0.720	0.871	0.453	0.712	0.891	0.961	0.960	0.876	-	0.412	0.862
oc	0.486	0.467	0.892	0.783	0.487	0.884	0.780	0.340	0.422	0.421	0.463	0.412	-	0.472
wic	0.674	0.844	0.493	0.734	0.887	0.527	0.763	0.803	0.866	0.864	0.930	0.862	0.472	-

Tabulka 4.3: Tabulka korelací pro hlavní komponentu. Zahrnuty i variace closeness.

4.3 Žebříčky významných autorů

Tabulky 4.4 a 4.5 ukazují prvních 30 autorů v pořadí podle metody h-index. Pro každého autora je zobrazeno jeho umístění podle všech zbylých implementovaných metod. Poslední řádek u každé metody označuje součet pořadí uvedených 30ti autorů.

Výstupy programu, tj. jednotlivá pořadí autorů a jejich ocenění jsou uvedeny v příloze A, v sekci A.1 pro databázi DBLP a v sekci A.2 pro databázi

	hi	ideg	odeg	deg	wideg	wodeg	wdeg	pr	btw	btwA	wBtwA
MICHAEL STONEBRAKER	1	1	7	1	1	6	2	2	2	2	1
DAVID J. DEWITT	2	2	5	2	2	3	1	14	3	3	2
JEFFREY D. ULLMAN	3	5	65	9	3	48	5	12	9	9	4
PHILIP A. BERNSTEIN	4	7	119	10	8	87	13	6	1	1	7
RAKESH AGRAWAL	5	14	3	5	10	9	7	40	27	24	19
WON KIM	6	6	39	6	11	12	8	21	11	12	42
CATRIEL BEERI	7	28	66	36	15	57	24	37	18	20	23
UMESHWAR DAYAL	8	10	36	8	16	39	16	27	4	5	47
SERGE ABITEBOUL	9	16	18	11	9	7	6	53	22	23	30
YEHOSHUA SAGIV	10	34	125	47	14	61	21	47	46	48	11
MICHAEL J. CAREY	11	9	4	4	5	1	3	36	12	8	5
CHRISTOS FALOUTSOS	12	36	15	22	18	11	10	87	101	90	53
NATHAN GOODMAN	13	20	71	27	23	47	27	23	16	16	15
JIM GRAY	14	3	279	7	4	256	11	3	6	4	3
JEFFREY F. NAUGHTON	15	25	35	24	22	23	17	70	67	58	22
HECTOR GARCIA-MOLINA	16	11	2	3	7	5	4	32	21	19	16
RONALD FAGIN	17	42	378	73	29	207	57	25	20	21	51
DAVID MAIER	18	12	84	15	12	82	23	31	13	14	14
HAMID PIRAHESH	19	22	33	17	24	19	15	57	33	28	20
RAGHU RAMAKRISHNAN	20	27	8	13	17	10	9	73	47	45	25
BRUCE G. LINDSAY	21	21	38	18	25	32	29	41	23	22	24
JENNIFER WIDOM	22	26	29	23	19	30	19	67	57	57	18
C. MOHAN	23	47	61	44	33	31	34	80	71	68	13
YANNIS E. IOANNIDIS	24	63	9	29	34	17	25	123	81	72	40
RAYMOND A. LORIE	25	4	854	14	6	550	20	5	8	6	6
SHAMKANT B. NAVATHE	26	30	12	16	44	25	35	54	42	49	154
RICHARD HULL	27	44	26	32	31	18	26	78	55	59	46
FRANCCEDILOIS BANCILHON	28	19	245	42	21	212	43	45	15	15	41
ARIE SHOSHANI	29	104	280	132	80	199	110	60	120	124	137
ALBERTO O. MENDELZON	30	68	34	43	43	40	38	81	90	95	43
	465	756	2980	733	586	2144	658	1330	1041	1017	923

Tabulka 4.4: Top 30 autorů DBLP podle H-indexu a pořadí podle ostatních metod

CiteSeer. Hodnoty u metody PageRank jsou v žebříčcích transformovány z intervalu $[0; 1]$ na $[0; |V|]$, protože při zaokrouhlení na tři desetinná místa jsou všechny hodnoty PageRanku zanedbatelně malé.

	hi	ideg	odeg	deg	wideg	wodeg	wdeg	pr	btw	btwA	wBtwA	ic	oc	wic
MICHAEL STONEBRAKER	1	1	7	1	1	6	2	2	2	2	1	1	24	1
AVID J. DEWITT	2	2	5	2	2	3	1	14	3	3	2	3	13	4
JEFFREY D. ULLMAN	3	5	65	9	3	48	5	12	9	9	4	5	58	5
PHILIP A. BERNSTEIN	4	7	119	10	8	87	13	6	1	1	7	6	217	7
RAKESH AGRAWAL	5	14	3	5	10	9	7	38	27	24	16	22	2	25
WON KIM	6	6	39	6	11	12	8	21	11	12	43	9	47	49
CATRIEL BEERI	7	28	66	36	15	57	24	35	18	20	22	24	41	17
UMESHWAR DAYAL	8	10	36	8	16	39	16	26	4	5	48	10	23	37
SERGE ABITEBOUL	9	16	18	11	9	7	6	50	22	23	32	39	14	51
YEHOSHUA SAGIV	10	34	125	47	14	61	21	44	46	48	11	38	97	14
MICHAEL J. CAREY	11	9	4	4	5	1	3	34	12	8	5	11	9	8
CHRISTOS FALOUTSOS	12	36	15	22	18	11	10	84	101	90	45	70	38	83
NATHAN GOODMAN	13	20	71	27	23	47	27	23	16	16	15	15	166	13
JIM GRAY	14	3	279	7	4	256	11	3	6	4	3	2	798	2
JEFFREY F. NAUGHTON	15	25	35	24	22	23	17	67	67	58	20	46	53	20
HECTOR GARCIA-MOLINA	16	11	2	3	7	5	4	31	21	19	18	18	4	15
RONALD FAGIN	17	42	378	73	29	207	57	25	20	21	50	23	701	29
DAVID MAIER	18	12	84	15	12	82	23	30	13	14	14	13	109	9
HAMID PIRAHESH	19	22	33	17	24	19	15	54	33	28	23	32	28	26
RAGHU RAMAKRISHNAN	20	27	8	13	17	10	9	70	47	45	24	52	11	22
BRUCE G. LINDSAY	21	21	38	18	25	32	29	39	23	22	25	20	50	18
JENNIFER WIDOM	22	26	29	23	19	30	19	64	57	57	19	55	19	21
C. MOHAN	23	47	61	44	33	31	34	77	71	68	13	67	116	27
YANNIS E. IOANNIDIS	24	63	9	29	34	17	25	117	81	72	35	85	6	40
RAYMOND A. LORIE	25	4	852	14	6	550	20	5	8	6	6	4	2024	6
SHAMKANT B. NAVATHE	26	30	12	16	44	25	35	51	42	49	150	34	8	327
RICHARD HULL	27	44	26	32	31	18	26	75	55	59	51	73	22	64
FRANCCEDILOIS BANCILHON	28	19	245	42	21	212	43	42	15	15	41	19	294	34
ARIE SHOSHANI	29	104	280	132	80	199	110	57	120	124	143	102	281	232
ALBERTO O. MENDELZON	30	68	34	43	43	40	38	78	90	95	42	90	30	33
	465	756	2978	733	586	2144	658	1274	1041	1017	923	988	5303	1239

Tabulka 4.5: Top 30 autorů největší komponenty DBLP podle H-indexu a pořadí podle ostatních metod

4.4 Porovnání metod s oceněními

[1]Platí pro největší silně spojitou komponentu

	Codd	ACM Fellows	Turing	ISI HC
hi	819	6 791 886	2 555 487	725 373
ideg	716	801 295	376 586	61 665
odeg	4 605	3 904 308	2 367 669	257 133
deg	741	1 077 282	525 324	84 191
wideg	571	802 148	374 344	61 757
wodeg	4 491	3 902 718	2 365 245	256 620
wdeg	692	1 065 801	511 523	82 652
pr	867	805 267	403 109	41 735
btw	796	828 571	399 720	70 342
btwA	821	1 024 010	428 133	68 996
wBtwA	567	1 057 788	473 146	77 168
ic[1]	805	63 753	34 004	1 384
oc[1]	6 607	129 810	59 701	4 098
wic[1]	512	73 729	38 045	1 482

Tabulka 4.6: Součty pořadí oceněných autorů

4.5 Aproximace betweenness centrality

Výpočet betweenness je časově nejnáročnější ze všech implementovaných metod. Pro citační síť autorů databáze DBLP není exaktní výpočet problém, ale pro rozsáhlou síť databáze CiteSeer je výpočet váženého betweenness časově velmi náročný úkol, jak můžeme nahlédnout jen z doby běhu aproximovaného betweenness v tabulce 4.1. Tabulka 4.7 znázorňuje Spearmanovo koeficienty korelace pořadí autorů DBLP mezi exaktním betweenness a aproximovanými, kde velikost množiny vrcholů, kterou uvažujeme ve výpočtu, je postupně $\frac{|V|}{2}, \frac{|V|}{4}, \frac{|V|}{8}, \ldots$.

zlomek velikosti množiny V	koeficient korelace
1	1.000 000
2	0.998 151
4	0.996 191
8	0.990 510
16	0.979 103
32	0.976 887
64	0.976 745

Tabulka 4.7: Tabulka Spearmano koeficientů korelace mezi exaktním a aproximovaným betweenness

5 Diskuse

5.1 Podobnost výsledků jednotlivých metod

5.2 Shoda výsledků s oceněními

5.3 Vliv vah na přesnost výsledků

5.4 Vstupní a výstupní hrany

6 Závěr

Literatura

[acm] Acm fellows. http://fellows.acm.org/.

[cit] Citeseerx. http://citeseerx.ist.psu.edu.

[DBL] Dblp bibliography. http://www.informatik.uni-trier.de/~ley/db.

[DPV08] S. Dasgupta, C. Papadimitriou, and U. Vazirani. *Algorithms*. McGraw-Hill, 2008.

[hig] Highly cited research. http://researchanalytics.thomsonreuters.com/highlycited/.

[LM06] Amy N. Langville and Carl D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, Princeton, NJ, 2006.

[sig] Sigmod awards. http://www.sigmod.org/sigmod-awards.

[tur] A.m. turing award. http://amturing.acm.org.

A Žebříčky významných autorů

A.1 DBLP

	Autor	hi	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	27.000		•	•	
2	DAVID J. DEWITT	25.000		•	•	
3	JEFFREY D. ULLMAN	24.000		•		
4	PHILIP A. BERNSTEIN	21.000		•		
5	RAKESH AGRAWAL	20.000		•		
6	WON KIM	20.000				
7	CATRIEL BEERI	20.000			•	•
8	UMESHWAR DAYAL	20.000		•		
9	SERGE ABITEBOUL	19.000		•	•	•
10	YEHOSHUA SAGIV	19.000				
11	MICHAEL J. CAREY	19.000		•		
12	CHRISTOS FALOUTSOS	19.000				
13	NATHAN GOODMAN	18.000				
14	JIM GRAY	18.000		•		
15	JEFFREY F. NAUGHTON	18.000				
16	HECTOR GARCIA-MOLINA	18.000		•		
17	RONALD FAGIN	18.000		•	•	•
18	DAVID MAIER	17.000		•		
19	HAMID PIRAHESH	17.000			•	
20	RAGHU RAMAKRISHNAN	17.000				
21	BRUCE G. LINDSAY	17.000				
22	JENNIFER WIDOM	17.000		•	•	
23	C. MOHAN	16.000		•		
24	YANNIS E. IOANNIDIS	16.000				
25	RAYMOND A. LORIE	16.000				
26	SHAMKANT B. NAVATHE	15.000				
27	RICHARD HULL	15.000			•	
28	FRANCCEDILOIS BANCILHON	15.000				
29	ARIE SHOSHANI	15.000				
30	ALBERTO O. MENDELZON	15.000				

	Autor	ideg	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	1909.000		•	•	
2	DAVID J. DEWITT	1484.000		•	•	
3	JIM GRAY	1400.000		•		
4	RAYMOND A. LORIE	1276.000				
5	JEFFREY D. ULLMAN	1180.000		•		
6	WON KIM	1146.000				
7	PHILIP A. BERNSTEIN	1145.000		•		
8	E. F. CODD	1110.000		•		
9	MICHAEL J. CAREY	1110.000		•		
10	UMESHWAR DAYAL	1076.000		•		
11	HECTOR GARCIA-MOLINA	1020.000		•		
12	DAVID MAIER	1017.000		•		
13	DONALD D. CHAMBERLIN	966.000		•	•	
14	RAKESH AGRAWAL	907.000		•		
15	PETER P. CHEN	906.000				
16	SERGE ABITEBOUL	848.000		•	•	•
17	KAPALI P. ESWARAN	847.000				
18	MORTON M. ASTRAHAN	846.000				
19	FRANCCEDILOIS BANCILHON	840.000				
20	NATHAN GOODMAN	819.000				
21	BRUCE G. LINDSAY	806.000				
22	HAMID PIRAHESH	803.000			•	
23	IRVING L. TRAIGER	785.000			•	
24	EUGENE WONG	762.000				
25	JEFFREY F. NAUGHTON	729.000				
26	JENNIFER WIDOM	727.000		•	•	
27	RAGHU RAMAKRISHNAN	724.000				
28	CATRIEL BEERI	722.000			•	•
29	NICK ROUSSOPOULOS	702.000				
30	SHAMKANT B. NAVATHE	694.000				

	Autor	**odeg**	**Turing**	**Codd**	**Fellows**	**ISI**
1	GERHARD WEIKUM	872.000			•	
2	HECTOR GARCIA-MOLINA	856.000		•		
3	RAKESH AGRAWAL	761.000		•		
4	MICHAEL J. CAREY	758.000		•		
5	DAVID J. DEWITT	758.000		•	•	
6	H. V. JAGADISH	717.000			•	
7	MICHAEL STONEBRAKER	677.000		•	•	
8	RAGHU RAMAKRISHNAN	652.000				
9	YANNIS E. IOANNIDIS	649.000				
10	ABRAHAM SILBERSCHATZ	636.000				
11	ELISA BERTINO	635.000			•	
12	SHAMKANT B. NAVATHE	629.000				
13	PHILIP S. YU	622.000				
14	STEFANO CERI	611.000				
15	CHRISTOS FALOUTSOS	607.000				
16	MATTHIAS JARKE	586.000				
17	GULTEKIN OUMLZSOYOGLU	582.000				
18	SERGE ABITEBOUL	575.000		•	•	•
19	NICK ROUSSOPOULOS	568.000				
20	MIRON LIVNY	559.000				
21	STANLEY Y. W. SU	558.000				
22	HANS-JOUMLRG SCHEK	557.000			•	
23	PATRICK VALDURIEZ	547.000				
24	GOETZ GRAEFE	546.000				
25	CLEMENT T. YU	542.000				
26	RICHARD HULL	537.000			•	
27	MICHAEL J. FRANKLIN	526.000				
28	RICHARD T. SNODGRASS	513.000			•	
29	JENNIFER WIDOM	510.000		•	•	
30	DENNIS SHASHA	508.000				

	Autor	deg	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	2586.000		•	•	
2	DAVID J. DEWITT	2242.000		•	•	
3	HECTOR GARCIA-MOLINA	1876.000		•		
4	MICHAEL J. CAREY	1868.000		•		
5	RAKESH AGRAWAL	1668.000		•		
6	WON KIM	1626.000				
7	JIM GRAY	1582.000		•		
8	UMESHWAR DAYAL	1562.000		•		
9	JEFFREY D. ULLMAN	1554.000		•		
10	PHILIP A. BERNSTEIN	1440.000		•		
11	SERGE ABITEBOUL	1423.000		•	•	•
12	H. V. JAGADISH	1411.000			•	
13	RAGHU RAMAKRISHNAN	1376.000				
14	RAYMOND A. LORIE	1364.000				
15	DAVID MAIER	1357.000		•		
16	SHAMKANT B. NAVATHE	1323.000				
17	HAMID PIRAHESH	1302.000			•	
18	BRUCE G. LINDSAY	1287.000				
19	ABRAHAM SILBERSCHATZ	1287.000				
20	GERHARD WEIKUM	1280.000			•	
21	NICK ROUSSOPOULOS	1270.000				
22	CHRISTOS FALOUTSOS	1264.000				
23	JENNIFER WIDOM	1237.000		•	•	
24	JEFFREY F. NAUGHTON	1216.000				
25	STEFANO CERI	1214.000				
26	PATRICK VALDURIEZ	1192.000				
27	NATHAN GOODMAN	1184.000				
28	DONALD D. CHAMBERLIN	1181.000		•	•	
29	YANNIS E. IOANNIDIS	1179.000				
30	MIRON LIVNY	1178.000				

	Autor	wideg	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	5946.000		•	•	
2	DAVID J. DEWITT	5733.000		•	•	
3	JEFFREY D. ULLMAN	4429.000		•		
4	JIM GRAY	3982.000		•		
5	MICHAEL J. CAREY	3583.000		•		
6	RAYMOND A. LORIE	3501.000				
7	HECTOR GARCIA-MOLINA	3275.000		•		
8	PHILIP A. BERNSTEIN	3225.000		•		
9	SERGE ABITEBOUL	3177.000		•	•	•
10	RAKESH AGRAWAL	3152.000		•		
11	WON KIM	2993.000				
12	DAVID MAIER	2772.000		•		
13	E. F. CODD	2736.000		•		
14	YEHOSHUA SAGIV	2575.000				
15	CATRIEL BEERI	2491.000			•	•
16	UMESHWAR DAYAL	2465.000		•		
17	RAGHU RAMAKRISHNAN	2426.000				
18	CHRISTOS FALOUTSOS	2413.000				
19	JENNIFER WIDOM	2354.000		•	•	
20	DONALD D. CHAMBERLIN	2269.000		•	•	
21	FRANCCEDILOIS BANCILHON	2264.000				
22	JEFFREY F. NAUGHTON	2186.000				
23	NATHAN GOODMAN	2176.000				
24	HAMID PIRAHESH	2135.000			•	
25	BRUCE G. LINDSAY	2013.000				
26	MORTON M. ASTRAHAN	1985.000				
27	IRVING L. TRAIGER	1820.000			•	
28	ABRAHAM SILBERSCHATZ	1791.000				
29	RONALD FAGIN	1773.000		•	•	•
30	EUGENE WONG	1764.000				

	Autor	wodeg	Turing	Codd	Fellows	ISI
1	MICHAEL J. CAREY	3239.000		•		
2	GERHARD WEIKUM	3071.000			•	
3	DAVID J. DEWITT	2818.000		•	•	
4	PHILIP S. YU	2614.000				
5	HECTOR GARCIA-MOLINA	2512.000		•		
6	MICHAEL STONEBRAKER	2316.000		•	•	
7	SERGE ABITEBOUL	2297.000		•	•	•
8	H. V. JAGADISH	2263.000			•	
9	RAKESH AGRAWAL	2240.000		•		
10	RAGHU RAMAKRISHNAN	2059.000				
11	CHRISTOS FALOUTSOS	2042.000				
12	WON KIM	1902.000				
13	ABRAHAM SILBERSCHATZ	1867.000				
14	MIRON LIVNY	1806.000				
15	GOETZ GRAEFE	1789.000				
16	STEFANO CERI	1775.000				
17	YANNIS E. IOANNIDIS	1775.000				
18	RICHARD HULL	1692.000			•	
19	HAMID PIRAHESH	1685.000			•	
20	HANS-JOUMLRG SCHEK	1661.000			•	
21	STANLEY Y. W. SU	1651.000				
22	CLEMENT T. YU	1630.000				
23	JEFFREY F. NAUGHTON	1587.000				
24	RICHARD T. SNODGRASS	1558.000			•	
25	SHAMKANT B. NAVATHE	1538.000				
26	ELISA BERTINO	1500.000			•	
27	ALON Y. LEVY	1487.000				
28	MICHAEL J. FRANKLIN	1454.000				
29	NICK ROUSSOPOULOS	1406.000				
30	JENNIFER WIDOM	1396.000		•	•	

	Autor	wdeg	Turing	Codd	Fellows	ISI
1	DAVID J. DEWITT	8551.000		•	•	
2	MICHAEL STONEBRAKER	8262.000		•	•	
3	MICHAEL J. CAREY	6822.000		•		
4	HECTOR GARCIA-MOLINA	5787.000		•		
5	JEFFREY D. ULLMAN	5643.000		•		
6	SERGE ABITEBOUL	5474.000		•	•	•
7	RAKESH AGRAWAL	5392.000		•		
8	WON KIM	4895.000				
9	RAGHU RAMAKRISHNAN	4485.000				
10	CHRISTOS FALOUTSOS	4455.000				
11	JIM GRAY	4374.000		•		
12	GERHARD WEIKUM	4193.000			•	
13	PHILIP A. BERNSTEIN	4091.000		•		
14	H. V. JAGADISH	3991.000			•	
15	HAMID PIRAHESH	3820.000			•	
16	UMESHWAR DAYAL	3778.000		•		
17	JEFFREY F. NAUGHTON	3773.000				
18	PHILIP S. YU	3765.000				
19	JENNIFER WIDOM	3750.000		•	•	
20	RAYMOND A. LORIE	3723.000				
21	YEHOSHUA SAGIV	3672.000				
22	ABRAHAM SILBERSCHATZ	3658.000				
23	DAVID MAIER	3657.000		•		
24	CATRIEL BEERI	3621.000			•	•
25	YANNIS E. IOANNIDIS	3470.000				
26	RICHARD HULL	3451.000			•	
27	NATHAN GOODMAN	3399.000				
28	MIRON LIVNY	3374.000				
29	BRUCE G. LINDSAY	3371.000				
30	GOETZ GRAEFE	3265.000				

	Autor	**pr**[1]	**Turing**	**Codd**	**Fellows**	**ISI**
1	E. F. CODD	179.324		•		
2	MICHAEL STONEBRAKER	137.371		•	•	
3	JIM GRAY	115.364		•		
4	DONALD D. CHAMBERLIN	114.010		•	•	
5	RAYMOND A. LORIE	107.204				
6	PHILIP A. BERNSTEIN	99.575		•		
7	MORTON M. ASTRAHAN	87.673				
8	KAPALI P. ESWARAN	87.167				
9	PETER P. CHEN	84.098				
10	IRVING L. TRAIGER	79.313			•	
11	JOHN MILES SMITH	78.833				
12	JEFFREY D. ULLMAN	74.323		•		
13	EUGENE WONG	68.319				
14	DAVID J. DEWITT	67.701		•	•	
15	MIKE W. BLASGEN	62.185				
16	GIANFRANCO R. PUTZOLU	61.585				
17	BRADFORD W. WADE	60.731				
18	RUDOLF BAYER	60.706		•		
19	JAMES W. MEHL	58.499				
20	PATRICIA P. GRIFFITHS	58.215				
21	WON KIM	57.946				
22	W. FRANK KING III	57.169				
23	NATHAN GOODMAN	56.791				
24	PAUL R. MCJONES	55.967			•	
25	RONALD FAGIN	54.766		•	•	•
26	RAYMOND F. BOYCE	54.475				
27	UMESHWAR DAYAL	54.099		•		
28	DIANE C. P. SMITH	53.677				
29	VERA WATSON	53.085				
30	MICHAEL HAMMER	52.687				

	Autor	**btw**	**Turing**	**Codd**	**Fellows**	**ISI**
1	PHILIP A. BERNSTEIN	62655703.293		•		
2	MICHAEL STONEBRAKER	61738362.921		•	•	
3	DAVID J. DEWITT	60335509.092		•	•	
4	JIM GRAY	58452724.132		•		
5	UMESHWAR DAYAL	58105048.655		•		
6	RAYMOND A. LORIE	57606842.228				
7	DONALD D. CHAMBERLIN	57435250.431		•	•	
8	MICHAEL J. CAREY	56191915.811		•		
9	JEFFREY D. ULLMAN	56098986.122		•		
10	KAPALI P. ESWARAN	55953909.624				
11	E. F. CODD	55595773.178		•		
12	WON KIM	55485910.707				
13	MORTON M. ASTRAHAN	53967137.730				
14	DAVID MAIER	53884993.441		•		
15	FRANCCEDILOIS BANCILHON	52436978.786				
16	NATHAN GOODMAN	51776071.388				
17	EUGENE WONG	50457002.386				
18	IRVING L. TRAIGER	50067735.663			•	
19	HECTOR GARCIA-MOLINA	49279794.248		•		
20	CATRIEL BEERI	49031169.516			•	•
21	RONALD FAGIN	48476621.189		•	•	•
22	BRUCE G. LINDSAY	47956637.448				
23	SERGE ABITEBOUL	47196023.670		•	•	•
24	RAKESH AGRAWAL	46621125.945		•		
25	PATRICIA G. SELINGER	45312957.343		•	•	
26	THOMAS G. PRICE	44961579.565				
27	DENNIS MCLEOD	44846630.893				
28	HAMID PIRAHESH	44408421.808			•	
29	HENRY F. KORTH	44365555.952			•	
30	RANDY H. KATZ	44264843.771				

	Autor	btwA	Turing	Codd	Fellows	ISI
1	PHILIP A. BERNSTEIN	159954377.951		•		
2	MICHAEL STONEBRAKER	154562285.765		•	•	
3	DAVID J. DEWITT	151226543.383		•	•	
4	UMESHWAR DAYAL	148199498.491		•		
5	DONALD D. CHAMBERLIN	146188880.058		•	•	
6	JIM GRAY	145464683.525		•		
7	E. F. CODD	144527193.507		•		
8	RAYMOND A. LORIE	143126778.632				
9	JEFFREY D. ULLMAN	142742025.957		•		
10	KAPALI P. ESWARAN	141829887.829				
11	WON KIM	141617051.174				
12	MICHAEL J. CAREY	139609158.987		•		
13	DAVID MAIER	136756549.030		•		
14	MORTON M. ASTRAHAN	135873540.410				
15	FRANCCEDILOIS BANCILHON	133327793.505				
16	NATHAN GOODMAN	131716143.987				
17	EUGENE WONG	127621249.542				
18	CATRIEL BEERI	126349161.825			•	•
19	IRVING L. TRAIGER	125392610.648			•	
20	RONALD FAGIN	124989300.213		•	•	•
21	HECTOR GARCIA-MOLINA	122564518.399		•		
22	SERGE ABITEBOUL	122057559.819		•	•	•
23	BRUCE G. LINDSAY	119996828.360				
24	DENNIS MCLEOD	118304803.594				
25	PETER P. CHEN	116123796.130				
26	JOHN MILES SMITH	115771141.296				
27	RAKESH AGRAWAL	115204381.342		•		
28	MICHAEL HAMMER	113451632.828				
29	NICK ROUSSOPOULOS	113024824.395				
30	PATRICIA G. SELINGER	112877895.367		•	•	

	Autor	wBtwA	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	52186391.843		•	•	
2	DAVID J. DEWITT	47843951.099		•	•	
3	JIM GRAY	46040049.553		•		
4	JEFFREY D. ULLMAN	43697973.877		•		
5	MICHAEL J. CAREY	40975341.551		•		
6	RAYMOND A. LORIE	38087173.986				
7	PHILIP A. BERNSTEIN	36384907.144		•		
8	LAWRENCE A. ROWE	35465224.669				
9	MIRON LIVNY	34770163.994				
10	EUGENE WONG	34705887.830				
11	YEHOSHUA SAGIV	33268899.535				
12	DONALD D. CHAMBERLIN	32864332.798		•	•	
13	C. MOHAN	32506688.148		•		
14	DAVID MAIER	32305917.497		•		
15	NATHAN GOODMAN	31533231.235				
16	HECTOR GARCIA-MOLINA	31173905.887		•		
17	RANDY H. KATZ	31141488.949				
18	JENNIFER WIDOM	30567077.674		•	•	
19	RAKESH AGRAWAL	30318090.387		•		
20	HAMID PIRAHESH	29476556.486			•	
21	E. F. CODD	29282563.027		•		
22	JEFFREY F. NAUGHTON	29221471.457				
23	CATRIEL BEERI	28703806.981			•	•
24	BRUCE G. LINDSAY	28493166.301				
25	RAGHU RAMAKRISHNAN	28228941.085				
26	GOETZ GRAEFE	26896490.365				
27	IRVING L. TRAIGER	26884342.802			•	
28	LAURA M. HAAS	26667276.137				
29	MORTON M. ASTRAHAN	26321142.833				
30	SERGE ABITEBOUL	24887257.714		•	•	•

	Autor	ic	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	0.593		•	•	
2	JIM GRAY	0.560		•		
3	DAVID J. DEWITT	0.556		•	•	
4	RAYMOND A. LORIE	0.556				
5	JEFFREY D. ULLMAN	0.546		•		
6	PHILIP A. BERNSTEIN	0.546		•		
7	E. F. CODD	0.543		•		
8	DONALD D. CHAMBERLIN	0.539		•	•	
9	WON KIM	0.537				
10	UMESHWAR DAYAL	0.535		•		
11	MICHAEL J. CAREY	0.532		•		
12	MORTON M. ASTRAHAN	0.531				
13	DAVID MAIER	0.529		•		
14	KAPALI P. ESWARAN	0.529				
15	NATHAN GOODMAN	0.527				
16	EUGENE WONG	0.526				
17	IRVING L. TRAIGER	0.525			•	
18	HECTOR GARCIA-MOLINA	0.523		•		
19	FRANCCEDILOIS BANCILHON	0.520				
20	BRUCE G. LINDSAY	0.519				
21	PETER P. CHEN	0.518				
22	RAKESH AGRAWAL	0.518		•		
23	RONALD FAGIN	0.517		•	•	•
24	CATRIEL BEERI	0.517			•	•
25	THOMAS G. PRICE	0.514				
26	PATRICIA G. SELINGER	0.514		•	•	
27	JOHN MILES SMITH	0.513				
28	MIKE W. BLASGEN	0.512				
29	RANDY H. KATZ	0.512				
30	GIO WIEDERHOLD	0.512				

	Autor	oc	Turing	Codd	Fellows	ISI
1	H. V. JAGADISH	0.475			•	
2	RAKESH AGRAWAL	0.473		•		
3	GERHARD WEIKUM	0.471			•	
4	HECTOR GARCIA-MOLINA	0.468		•		
5	GULTEKIN OUMLZSOYOGLU	0.467				
6	YANNIS E. IOANNIDIS	0.467				
7	STEFANO CERI	0.466				
8	SHAMKANT B. NAVATHE	0.466				
9	MICHAEL J. CAREY	0.465		•		
10	ELISA BERTINO	0.464			•	
11	RAGHU RAMAKRISHNAN	0.463				
12	RICHARD T. SNODGRASS	0.463			•	
13	DAVID J. DEWITT	0.462		•	•	
14	SERGE ABITEBOUL	0.462		•	•	•
15	CLEMENT T. YU	0.460				
16	GOETZ GRAEFE	0.460				
17	HANS-JOUMLRG SCHEK	0.459			•	
18	ABRAHAM SILBERSCHATZ	0.459				
19	JENNIFER WIDOM	0.459		•	•	
20	PATRICK VALDURIEZ	0.458				
21	NICK ROUSSOPOULOS	0.457				
22	RICHARD HULL	0.457			•	
23	UMESHWAR DAYAL	0.457		•		
24	MICHAEL STONEBRAKER	0.454		•	•	
25	DENNIS SHASHA	0.454				
26	MATTHIAS JARKE	0.453				
27	MIRON LIVNY	0.451				
28	HAMID PIRAHESH	0.451			•	
29	CHRISTIAN S. JENSEN	0.451				
30	ALBERTO O. MENDELZON	0.450				

	Autor	wic	Turing	Codd	Fellows	ISI
1	MICHAEL STONEBRAKER	2.055		•	•	
2	JIM GRAY	2.047		•		
3	E. F. CODD	2.043		•		
4	DAVID J. DEWITT	2.017		•	•	
5	JEFFREY D. ULLMAN	2.014		•		
6	RAYMOND A. LORIE	2.012				
7	PHILIP A. BERNSTEIN	2.011		•		
8	MICHAEL J. CAREY	1.993		•		
9	DAVID MAIER	1.979		•		
10	EUGENE WONG	1.973				
11	DONALD D. CHAMBERLIN	1.969		•	•	
12	LAWRENCE A. ROWE	1.967				
13	NATHAN GOODMAN	1.966				
14	YEHOSHUA SAGIV	1.966				
15	HECTOR GARCIA-MOLINA	1.960		•		
16	IRVING L. TRAIGER	1.955			•	
17	CATRIEL BEERI	1.949			•	•
18	BRUCE G. LINDSAY	1.944				
19	MORTON M. ASTRAHAN	1.943				
20	JEFFREY F. NAUGHTON	1.942				
21	JENNIFER WIDOM	1.937		•	•	
22	RAGHU RAMAKRISHNAN	1.936				
23	MIRON LIVNY	1.936				
24	RANDY H. KATZ	1.934				
25	RAKESH AGRAWAL	1.933		•		
26	HAMID PIRAHESH	1.933			•	
27	C. MOHAN	1.921		•		
28	DONOVAN A. SCHNEIDER	1.910				
29	RONALD FAGIN	1.910		•	•	•
30	LAURA M. HAAS	1.906				

A.2 CiteSeer