Edícia výskumných textov
informatiky a informačných technológií

Weboveda:
východiská, predmet, metódy

Pavol Návrat, Peter Kubán, Peter Krátky, Peter Macko,
Róbert Móro, Ivan Srba, Márius Šajgalík, Jakub Ševcech,
Petra Vrablecová

Weboveda:
východiská, predmet, metódy

Slovenská technická univerzita v Bratislave
2014

Edícia výskumných textov informatiky a informačných technológií

Weboveda: východiská, predmet, metódy
Štúdie vybraných tém programových a informačných systémov (5)

Posudzovatelia: doc. RNDr. Michal Laclavík, PhD.
doc. Ing. Kristína Machová, PhD.

Návrh grafickej úpravy: Ing. Peter Macko, prof. Ing. Mária Bieliková, PhD.
Technický redaktor: Ing. Peter Macko
Technická spolupráca: Ing. Peter Krátky
Grafika na obálke: Mgr. Alena Kovárová, PhD.
Návrh obálky: Ing. Peter Kaminský

Kniha vznikla a bola vydaná s finančnou podporou projektu Agentúry na podporu výskumu a vývoja:

- APVV-0208-10 Kognitívne cestovanie po digitálnom svete webu a knižníc s podporou personalizovaných služieb a sociálnych sietí

a s podporou projektov Vedeckej grantovej agentúry Ministerstva školstva Slovenskej republiky a Slovenskej akadémie vied (VEGA):

- VG 1/0675/11 Kontextové vyhľadávanie a prehliadanie informácií v sociálnom prostredí webu
- VG 1/0752/14. Inteligentná analýza veľkých údajových korpusov sémanticky-orientovanými a bio-inšpirovanými metódami v paralelnom prostredí.

Publikáciu podporili: GRATEX IT INŠTITÚT v rámci fondu GraFIIT.

Schválilo vedenie Fakulty informatiky a informačných technológií STU v Bratislave dňa 1.4.2014, uznesenie číslo 1/01-04-2014 v edícii výskumných textov.

Fakulta informatiky a informačných technológií Slovenskej technickej univerzity v Bratislave
Ilkovičova 2, 842 16 Bratislava
http://www.fiit.stuba.sk/, pavol.navrat@stuba.sk

Vydala Slovenská technická univerzita v Bratislave
v Nakladateľstve STU, Bratislava, Vazovova 5.

ISBN online 978-80-227-4264-1

PREDHOVOR

Táto knižka je pokusom opísať pokus. Pokúšame sa opísať vo svete práve prebiehajúci pokus zaviesť novú vednú disciplínu. Web je tu a niektorí sa domnievajú, že je to tak dôležitý predmet skúmania s tak osobitnými vedeckými metódami, že je rozumné zaviesť (a rozvíjať a pestovať) zvláštnu vedenú disciplínu, ktorá sa tomu bude venovať. Podobne zmýšľajúci nadšenci sa stretli v septembri 2005 v Londýne na dvojdňovej tvorivej dielni už nazvanej Web Science Workshop. Názov Web Science je trochu provokujúci a kontroverzný aj v angličtine. Web sa stáva predmetom skúmania a štúdia podobne ako príroda (alebo jej určitá časť či aspekt), človek alebo spoločnosť. Web je však aj predmetom navrhovania a treba ho spravovať a prevádzkovať, čo si vyžaduje inžinierstvo. Web sa dnes skúma vo viacerých disciplínach. Ukazuje sa však, že to na jeho celostné pochopenie a rozvíjanie nemusí stačiť. Treba interdisciplinárny prístup. Podporovatelia myšlienky zrodu novej vednej disciplíny v nej vidia najlepšiu možnosť pre takýto prístup. Fakt, že už vznikli vedecké časopisy a usporadúvajú sa vedecké konferencie, venované Web Science, je v tomto smere sľubný. Či sa však naozaj nová vedná disciplína definitívne etabluje vo vedeckom priestore, je otázkou budúcnosti. Medzitým Web Science prenikla do vzdelávacieho priestoru a už sú univerzity, ktoré ponúkajú študijné programy s týmto názvom.

Táto knižka je výsledkom doktorandského seminára, ktorý som viedol v akademickom roku 2013/2014. Na Fakulte informatiky a informačných technológií máme šťastie na šikovných premýšľajúcich študentov. Neboja sa výziev. Úloha opísať v sérii seminárnych príspevkov rodiacu sa vednú disciplínu, s ktorou som prišiel na začiatku seminára, bola celkom slušnou výzvou. Náročné bolo už len hľadať vedecké pramene vzhľadom na novosť tématiky. Pri návrhu tém jednotlivých seminárnych stretnutí sme sa nechali inšpirovať McCownovým sylabusom pre úvod do webovedy [1, 2], keďže predstavuje celkom dobrý vyvážený pohľad na jednotlivé obsahové témy, spadajúce do alebo súvisiace s webovedou. S vďakou uvádzame, že aj pri písaní viacerých kapitol sme vychádzali z obsahu inšpiratívnych McCownových prednášok.

Jednou z pretrvávajúcich otázok nášho seminára bolo hľadanie slovenského názvu pre novú vednú disciplínu. Otáznikov je pritom viac. Zdá sa, že slovo web sa v slovenčine už udomácnilo, aj keď sa začala používať aj pavučina a nie je jasné, čo sme prebratím cudzieho slova získali. Návrh na pavučinológiu bol však týmto znevýhodnený. Návrh na sieťopis mal zasa tú nevýhodu, že sieť je širší pojem ako web. Po viacerých diskusiách sme sa ustálili na webovede. Uvedomujeme si, že ide o novotvar. Môže vyvolávať rôzne reakcie, ale to už patrí k veci.

Po prednesení príspevkov a diskusii na seminári spracovali autori témy aj písomne. Prvotnú zodpovednosť za kapitoly sme si podelili takto: Návrat za kapitoly 1, 16, Kubán za kapitoly 9, 14, Krátky za kapitoly 5, 10, Macko za kapitoly 2, 8, Móro za kapitoly 6, 13, Srba za kapitolu 12, Šajgalík za kapitolu 7, Ševcech za kapitoly 3, 10, 15, Vrablecová za kapitoly 4, 11. Písanie

som koordinoval a texty som aj (trochu) redakčne upravoval. Vytvorenie konečnej podoby si vyžadovalo aj mnoho ďalšej práce technického charakteru, podobne ako aj fungovanie seminára. Účastníci seminára si zaslúžia poďakovanie nielen za príspevky, ale aj za samosprávne fungovanie seminára a technickej prípravy tejto knižky. Obzvlášť chcem poďakovať Petrovi Mackovi a Petrovi Krátkemu.

O návrh titulnej grafiky som požiadal A. Kovárovú. Za návrh aj za jej ochotu, s akou k veci pristúpila, jej veľmi ďakujem. Obrázok O´Haru a Hallovej [3], ilustrujúci interdisciplinárnosť webovedy je nielen výstižný, ale aj inšpirujúci. Ďakujeme autorom za inšpiráciu pre grafiku na obálke.

Dúfam, že táto knižka bude užitočná pre niekoho, kto sa zaujíma o web ako fenomén významne ovplyvňujúci náš život a poslúži ako východisko pre jeho ďalšie hlbšie štúdium.

V Bratislave, jún 2014 Pavol Návrat

Literatúra

[1] McCown, F., Nelson, M.L.: Resources for teaching web science to computer science undergraduates (abstract only). In *Proceedings of the 45th ACM technical symposium on Computer science education (SIGCSE '14)*, (2014), pp. 725-725.

[2] McCown, F.: Introduction to Web Science. Syllabus, (2013). Dostupné na: http://www.cs.odu.edu/~mln/teaching/cs595-f13/?method=display&element=~week-01&metadata=descriptive

[3] O'Hara, K., Hall, W.: Web Science, ALT Online Newsletter, (2008)

PREDHOVOR K ELEKTRONICKÉMU VYDANIU

Keď sme sa s doktorandami – spoluautormi rozhodli, že sa vytrápime a pripravíme naše spracovanie zvolenej témy do tlače tak, aby mohla z toho vzniknúť knižka, robili sme to pre radosť z poznania, o ktoré sme sa chceli podeliť aj s inými. Prvé reakcie na toto skromné dielko sú povzbudivé v tom, že sú ľudia, ktorí si v ňom radi zalistujú. Aby tých ľudí mohlo byť čo najviac, využívame možnosť zverejniť ho v tejto nepatrne obmenenej podobe aj online. Za túto možnosť ďakujem Nakladateľstvu STU.

V Bratislave, október 2014 Pavol Návrat

.

OBSAH

1 Východiská

Uvedieme aspoň niektoré technologické a spoločenské východiská pre rozvoj a štúdium webovedy. Medzi technologickými východiskami spomenieme internet, systém doménových mien, jednotný identikátor zdroja, hypertext. Medzi spoločenskými východiskami výskumu spomenieme sociálne vzťahy, podporené službami ako sú blog, wiki, sídlo sociálneho zosieťovania, mikroblog.

1.1 Technologické východiská

Web je tu. Síce stále ešte menej než štvrť storočia, ale dokázal už zásadným spôsobom zmeniť alebo ovplyvniť náš život. A nezdá sa (zatiaľ), že by boli v dohľade hranice jeho ďalšieho rozvoja. Vedecký a technický pokrok prináša zmeny, ktoré si azda nikto ani nevedel predstaviť.

Základným vynálezom, ktorý otvoril cestu ďalších prevratných zmien, je počítač. Počítač je vo svojich fundamentálnych princípoch aj vo fyzikálnej (elektronickej) báze jeho realizácie stále viac menej rovnaký. Je neporovnateľne rýchlejší než bol voľakedy a každý ďalší rok stále porovnateľne (podľa Moora zhruba dvojnásobne) rýchlejší než bol pred jeden a pol či dvoma rokmi. Podobne exponenciálne sa zvyšuje aj veľkosť jeho pamäti.

Počítače sa sprvoti využívali najmä na hromadné spracovanie údajov a na vedecké výpočty. Počítače boli veľké (fyzicky) a veľa stáli. Vlastnili ich najmä podniky, výskumné inštitúcie a vládne inštitúcie. Postupné pochopenie možností, ktoré počítačové spracovanie informácií prináša, viedlo k uvedomeniu si významu prepojenia viacerých počítačov. Návrh protokolov pre odovzdávanie si paketov údajov medzi počítačmi umožnil vznik počítačových sietí.

Technologický pokrok prinášal neustálu miniaturizáciu (integrované obvody, mikroprocesory), avšak boli to časy, keď by myšlienku, že nejaká súkromná osoba vlastní svoj počítač, považovali skoro za absurdnú. K prielomu prispeli „garážoví“ inovátori, ktorí dokázali zostrojiť mikropočítač, ktorý nezaberá celú miestnosť, ale dá sa postaviť na dosku stola a dokonca je aj lacnejší. Vznikol osobný počítač s klávesnicou a obrazovkou, ktorý nebol určený na to, aby ho používala organizácia, ale jedinec.

Otvoril sa potenciálne obrovský trh zákazníkov, ktorí sa mohli stať vlastníkmi počítača. Čím viac sa darilo osobných počítačov predávať, tým mohla byť ich cena nižšia, čo naspäť podporilo ich predaj. Nebolo to vôbec také samozrejmé. To naozaj potrebujem ja, bežný človek, robiť toľko výpočtov, aby sa mi oplatilo kúpiť také (stále ešte relatívne drahé) zariadenie, zvané osobný počítač? Aspoň že „hrozbu", že by si každý musel svoj počítač aj programovať, čiastočne eleminovali balíky programov a v nich napr. tabuľkové procesory, v ktorých si mohol človek rátať rodinný rozpočet. Najmä však vznikali a šírili sa počítačové hry. Ich použitie nielen sprí-jemnilo človeku toto technické zariadenie, ale znamenalo prielom do toho, na čo sa dá počítač použiť. Počítač definitívne prestal byť zariadením (len) na počítanie.

Medzitým sa rozvíjali a rozširovali aj počítačové siete. Prepojením viacerých počítačových sietí vznikol internet. Superpočítače v nich spojené dokázali utiahnuť veľké množstvá terminálov, čo boli miesta pre individuálne použitie počítača. Zrazu sa veľký počet ľudí, potenciálne používajúcich počítač prostredníctvom terminálu, prepojil do jednej siete, ak boli tie počítače prepojené. Počítačové prepojenie umožňovalo, aby si medzi sebou posielali správy. E-pošta sa stala zabijáckou aplikáciou internetu. Ľudia zistili, že s niekým na druhej strane zemegule si možno za jediný deň vymeniť aj tucet listov. Dôsledky takéhoto zefektívnenia komunikácie na rozvoj globálneho obchodu a podnikania sú prevratné.

Internet neumožnil len e-poštu. Pomocou protokolu FTP sa dal stiahnuť dokument z iného, ľubovoľne vzdialeného počítača. Toto všetko sa nezaobíde bez nejakého spôsobu identifikovania počítača a osoby. Systém doménových mien dáva mená počítačom, službám alebo akýmkoľvek zdrojom pripojeným do počítačovej siete (internetu). Navrhli ho hierarchicky. Priestor doménových mien tvorí strom. V identifikácii počítača, poskytujúceho nejaké služby (servera) stuba.sk je sk meno vrchnej domény a stuba je jedna z poddomén domény sk. Takáto identifikácia je pre človeka zrozumiteľnejšia a zapamätateľnejšia než skutočná adresa počítača 147.175.1.18, ktorou sa riadi komunikácia v sieti podľa internetového protokolu. Preklad z mien do adries robia automaticky servery doménových mien. Mimochodom, tie vedia popri tom aj spracovať záznamy, ktoré pre príslušnú doménu určujú meno webového servera a preto napríklad netreba pri sprístupňovaní webu písať www.stuba.sk. Aj e-adresy používajú tento spôsob na identifikáciu poštového servera. Pred neho sa píše meno adresáta a znak @, napríklad pavol.navrat@stuba.sk. Poštový server je tu webmail.stuba.sk, ale vďaka záznamu v systéme doménových mien sa píše len stuba.sk.

Aj keď protokol FTP umožňuje, že si môže človek stiahnuť (na stiahnutie určený) dokument z hociktorého počítača na Zemi, pokiaľ je pripojený do internetu, čo bol samo osebe obrovský pokrok, pre sťahujúceho človeka to nebolo veľmi pružné. Sťahoval dokument, ktorého obsah nevidel. Ale scéna bola pripravená pre zásadný prielom.

Zárodočným jadrom pre prielom sa stal CERN, európska organizácia pre jadrový výskum. Má výlučné postavenie ohľadne finančnej podpory, ktorá umožňuje účasť obrovského počtu výskumníkov z mnohých krajín. V roku 1989 medzi nimi pôsobil aj softvérový inžinier Tim Berners-Lee. Chápal, že toto množstvo výskumníkov si potrebuje vymieňať dokumenty, údaje, dokonca aj softvér a to aj po tom, ako už nie sú fyzicky v CERNe. Prišiel s víziou pavučiny

(webu) dokumentov, ktoré môžu obsahovať odkazy na iné dokumenty. Spomenul si na pojem hypertextu ako textu, ktorý je čitateľný pre človeka a obsahuje spojenia na iné také texty. Navrhol jazyk HTML, v ktorom sa publikujú dokumenty na webe. Navrhol protokol HTTP, ktorý opisuje, ako sa sprístupňujú prepojené dokumenty. Tretím návrhom je spôsob, ako jednotným spôsobom identifikovať (pomenovať aj adresovať) zdroje na webe.

Výsledkom je jednotný priestor dokumentov, ktoré môžu byť ľubovoľne navzájom prepojené. Priestor zahŕňa celý svet (preto celosvetová pavučina) a zahrnutie dokumentu do neho nie je podmienené jeho geografickou polohou. Prepojenia pripomínajú bibliografické odkazy, ale sú "živé". Bernes-Lee naprogramoval aj prvý webový prehliadač a webový server. Umožnil, aby sa v prehliadaní dalo plynulo pokračovať v odkazovanom dokumente, čo je sen každého čitateľa textu, obsahujúceho odkazy na literatúru. Namiesto vyhľadávania odkazovaných zdrojov po knižiniciach stačí jednoduché kliknutie na hyperodkaz. Webový prehliadač so serverom zariadia zvyšok.

To už bol rok 1993. CERN dal vyvinutú webovú technológiu k dispozícii verejnosti. Webové stránky začali pribúdať aj mimo CERNu. Dnes web používa veľká časť obyvateľov Zeme a počet webových stránok stále rastie. Č je však asi dôležitejšie, pokračuje vývoj webu samotného – od adaptívneho k sémantickému [6], od 2.0 k 3.0, od ontológií k folksonómiám, cez personalizáciu ku grupizácii atď. [2, 3, 4, 8, 9]. Zásadným spôsobom sa zmenilo, kde a ako vyhľadávame informácie [5, 11].

1.2 Spoločenské východiská

S webom sú tu aj nové alebo zmenené už jestvujúce spôsoby medziľudskej a sociálnej komunikácie aj formy medziľudských a sociálnych vzťahov. Web nie je len akýmsi dômyselným nástrojom, ktorý uľahčuje komunikáciu alebo skvalitňuje vzťahy. Web do nich prináša novú kvalitu a spoluvytvára ich.

Tak, ako sa rozšírilo používanie e-pošty, narastalo aj množstvo správ, z ktorých mnohé obsahovali zaujímavé informácie nielen pre pôvodného adresáta správy. Vznikli tématické skupiny, venované rôznym témam. Niekto tému začal napríklad otázkou a nasledovali reakcie iných, ktorí mali a chceli k tej téme niečo napísať. Usenet je príkladom systému, ktorý uchováva také množiny správ. Podobne môžu fungovať diskusné skupiny na webe. O nejakej téme diskutujú ľudia, ktorí sa vôbec nemusia poznať. Záznam ich diskusie môžu neskôr čítať ľudia, ktorí sa pôvodnej diskusie vôbec nezúčastnili. V diskusii o nejakej téme sa mohli napísať aj užitočné rady, ktoré sú zaujímavé pre neskoršieho čitateľa.

Webové sídlo sa nestalo výsadou organizácií či úradov, aj keď pre jednotlivcov nebolo pôvodne najmä s ohľadom na svoju statickosť veľmi atraktívne. Je však dosť ľudí, ktorým stačí, že ich texty ľudia čítajú. Na webové sídlo začali písať svoje zápisky či komentáre na zvolené témy. Takýto v podstate akýsi webový denník či záznamník (weblog alebo skrátene blog) si často našiel svojich čitateľov. Technologická inovácia (RSS) im uľahčila sledovanie obľúbených blogerov v tom, že po objednaní tejto služby dostanú automaticky správu o tom, že pribudol nový blog. Tak, ako pribúdali interaktívne možnosti webu, mohli blogeri pod svoje blogy pridať

aj pozvánku na diskusiu. Niekedy sú diskusie tak ohnivé, že prekročia hranice slušnosti a čo je horšie, aj etnickej alebo náboženskej znášanlivosti. Kto nesie zodpovednosť za zverejňovanie takýchto názorov? Je prípustné alebo naopak želateľné ich mazať (cenzurovať)?

Webové sídlo môže byť ešte interaktívnejšie. Wiki je webové sídlo, ktorého obsah môžu spoločne vytvárať viacerí ľudia. Podporuje spoluprácu ľudí, ktorí vôbec nemusia byť technicky zdatní.

Sociálna sieť je pôvodne pojem, ktorý zaviedli v sociálnych vedách pri štúdiu vzťahov medzi jednotlivcami, skupinami, organizáciami alebo dokonca spoločnosťami. Opisuje sociálnu štruktúru, určenú ich interakciami. Študoval sa už v prvej polovici dvadsiateho storočia, keď sa ešte o počítačoch a tobôž o webe nechyrovalo. Zvyšujúca sa možnosť interaktívnosti webových sídel priniesla nápad podporiť interakciu medzi ľuďmi poskytnutím webového sídla, ktoré umožňuje komukoľvek vytvoriť si akýsi osobný profil, prehlásiť sa za „priateľa“ inej osoby so zverejneným profilom alebo pridať sa do jednej či viacerých skupín. Momentálne je najpopulárnejšou realizáciou takého nápadu sídlo alebo služba sociálneho zosieťovania, nazývaná Facebook.

Vývoj posledných rokov priniesol aj podporu písania a šírenia tzv. mikroblogov, t.j. kratučkých poznámok, ktoré nemajú viac než 140 znakov.

Tieto aj ďalšie nové možnosti sociálnej komunikácie alebo vzťahov v spojitosti s webom sú novými javmi. Stávajú sa pomaly predmetom skúmania sociálnych vied [1], ale skúmanie sa nezaobíde bez hlbokého pochopenia vlastností webu. Na druhej strane, ich pochopenie je nevyhnutné pre hlbšie pochopenie toho, čo je web. Málokto asi bude vážne tvrdiť, že označenie niekoho za „priateľa“ v službe sociálneho zosieťovania robí z týchto dvoch ľudí priateľov. Ak však odhliadneme od tohto označenia, nejaká sociálna väzba medzi dvoma ľuďmi vzniká. Je príkladom špecifického sociálneho vzťahu, ktorý je podmienený existenciou webu. Ďaleko viac, hypotéza webovedy znie, že poznanie vlastností aj takýchto sociálnych väzieb vznikajúcich v prostredí webu je potrebné pre jeho lepšie pochopenie.

1.3 Zhrnutie

Uviedli sme niektoré dôležité technologické a spoločenské východiská, ktoré významným spôsobom určujú vývoj webu ako úplne nového technologického a spoločenského javu. Tento jav treba skúmať, aby sme lepšie poznali jeho podstatu a jeho vlastnosti. Treba ho ďalej rozvíjať, budovať a prevádzkovať, aby lepšie slúžil rastúcim potrebám ľudí. Nezaškodí spomenúť, že web má veľký potenciál ovplyvniť vývoj iných disciplín [10]. Jeho komplexné štúdium si nevyhnutne vyžaduje interdisciplinárny prístup. Možno si vyžaduje aj novú vednú disciplínu – webovedu.

Literatúra

[1] Ackland, R.: *Web social science: Concepts, Data and Tools for Social Scientists in the Digital Age*. Sage Publications Ltd., (2013).

[2] Berners-Lee, T., Hall, W., Hendler, J., Shadbolt, N., and Wietzner, D.: Creating a science of the Web. In *Science*, vol. 313, no. 5788, (2006), pp. 769-771.

[3] Berners-Lee, T., Hall, W., Hendler, J., O'Hara, K., Shadbolt, N., and Weitzner, D.: A framework for Web science. In *Foundations and Trends in Web Science 1*, (2006).

[4] Hendler, J., Shadbolt, N., Hall, W., Berners-Lee, T. and Weitzner, D.: Web science: an interdisciplinary approach to understanding the web. In *Communications of ACM*, vol. 51, issue 7, (2008), pp. 60-69.

[5] Laclavík, M. a Šeleng, M.: *Vyhľadávanie informácií*. Nakladateľstvo STU, Bratislava, 2012.

[6] Machová, K.: *Od adaptívneho k sémantickému webu*. Technická univerzita v Košiciach, Košice, 2013.

[7] McCown, F., Nelson, M.L.: Resources for teaching web science to computer science undergraduates (abstract only). In *Proceedings of the 45th ACM technical symposium on Computer science education* (*SIGCSE '14)*, (2014), pp. 725-725.

[8] McCown, F.: Introduction to Web Science. Syllabus, (2013). Dostupné na: http://www.cs.odu.edu/~mln/teaching/cs595-f13/?method=display&element=~week-01&metadata=descriptive

[9] Shneiderman, B.: Web science: a provocative invitation to computer science. In *Communications of ACM*, vol. 50, no. 6 (2007), pp. 25-27.

[10] Bieliková, M., Návrat, P., Chudá, D., Polášek, I., Barla , M., Tvarožek, J., Tvarožek, M.: Webification of Software Development: General Outline and the Case of Enterprise Application Development. In *AWERProcedia Information Technology & Computer Science*, vol. 3, pp. 1157-1162

[11] Návrat, P., Kovaľ, R.: Intelligent Support for Information Retrieval of Web Documents. In *Computing and Informatics*, vol. 21, issue 5, (2002), pp. 509-528.

2 Základy webu

Web je jedným z ľudských výtvorov, ale aj napriek tomu je tajomnou a neprebádanou oblasťou. Dnešný web sa skladá z obrovského množstva uzlov a prepojení a tvorí tak jednu z najväčších grafových sústav. Web si v nedávnej minulosti získal veľkú popularitu a tak sa táto oblasť stala veľmi zaujímavou nielen pre výskumníkov, ale aj pre bežných používateľov. Vďaka tomu sa na webe nachádza obrovské množstvo informácií. Sú však publikované v neštruktúrovanej forme a je nutné hľadať cesty ich spracovania a získavania pridanej hodnoty z týchto údajov. Okrem toho prišiel web aj s novým sociálnym rozmerom, ktorý prináša do oblasti webu veľké výhody, ale aj radu nevýhod v podobe straty súkromia.

V dnešnej dobe sa web stáva neoddeliteľnou súčasťou ľudského života. V počiatkoch bol počítač vecou, ktorú vedeli využívať iba skutoční nadšenci. Postupne sa však počítače stále viac udomácňovali v domovoch bežných ľudí. S príchodom webu sa tento rozmach dostáva nielen do domácnosti, ale aj do vreciek používateľov v podobe prenosných (mobilných) zariadení ako sú tzv. mobily (v minulosti známe aj ako mobilné telefóny). Obrovský rozmach webu je vidieť najmä od príchodu „webu 2.0“. Web od tohto momentu získal sociálny efekt, čo ho zakorenilo ešte hlbšie do ľudských životov. Dnes je absolútne bežné, že sú ľudia pripojení na internet zo svojich mobilov, zverejňujú fotografie alebo videá z odľahlých častí Zeme alebo zisťujú aktuálnu dopravnú situáciu priamo na cestách z auta.

Veľkosť dnešného webu sa preto stále zväčšuje. Webové giganty ako YouTube, Google alebo Facebook nám prinášajú štatistiky ohromného nárastu ich databáz. Napríklad na server s videami YouTube pribudne počas minúty *tridsať hodín* videa. Pomocou vyhľadávača Google sa zrealizujú *dva milióny* vyhľadávaní [1]. Štatistiky z roku 2012 taktiež hovoria o tom, že počet zariadení pripojených na internet bol v tomto roku zhruba rovnaký, ako je populácia Zeme. Odhad predpokladá, že v roku 2015 bude počet zariadení pripojených na internet zhruba dvojnásobkom počtu ľudskej populácie.

Takéto rozšírenie internetu má na svedomí aj minutie IP adries verzie 4, pri zavedení ktorých si málokto dokázal predstaviť, že tento rozsah sa niekedy minie. Dnes pri zavedení IP verzie 6 si taktiež vieme len ťažko predstaviť, že by sme minuli 2^{128}, čo je asi 3,402823669 x 10^{38} jedinečných adries [2]. Táto doba však môže prísť skôr než sa nazdáme v prípade, že bude mať pripojenie na internet každé zariadenie v domácnosti, ako pračka, mikrovlnná rúra, chladnička a podobne. Všeobecnejšie sa táto predstava nemusí obmedziť len na domácnosť. Hovorí sa o internete vecí.

Takýto enormný nárast používateľov a údajov uložených na rôznych serveroch je však veľmi ťažké efektívne spracovať. Tomuto problému sa venuje veľa výskumu a veľa výskumníkov tejto téme zasväcuje svoj život. Práve z týchto dôvodov začína byť potrebné zaoberať sa potrebou definovania novej vednej disciplíny, disciplíny, ktorá sa bude venovať webu, webovým technológiám a faktorom ovplyvňujúcich web.

Sir Nigel Shadbolt

1956

Profesor umelej inteligencie na southampton-skej univerzite. Vedie výskum v oblasti prepojených údajov, sémantického webu, expertných systémov...

Sir Tim Berners-Lee

1955

Vynálezca Webu, ktorý vytvoril prvú webovú stránku 6. augusta 1991. Zaoberá sa sémantickým webom ktorý učí na Southampton-skej univerzite.

Wendy Hall

1952

Viedla tím, ktorý vyvinul Microcosm hypermedia, ešte pred WWW (systém pre hypermedia). Pôsobí na Southamptonskej univerzite, kde sa zaoberá výskumom multimedií a hipermédií.

James Hendler

1957

Výskumník zaoberajúci sa sémantickým webom a umelou inteligenciou. Študoval na Brown-ovej univerzite. Je súčasťou tímu, ktorý vyvíja Watson RPI.

2.1 Základy novej vednej disciplíny

Weboveda, ako nová vedná disciplína využíva poznatky z množstva iných vedných disciplín. Táto veda sa zaoberá štúdiom webu ako celku a fenoménu. To zahŕňa štúdium vlastností webu, protokolov, algoritmov a sociálnych efektov.

Základy tejto vednej disciplíny možno hľadať v roku 2006. V tomto roku vznikla Webovedná výskumná iniciatíva (Web Science Research Initiative, WSRI) ako výsledok spolupráce medzi MIT CSAIL a univerzitou v Southamtone. Tieto dve univerzity sa chceli pomocou tejto vednej disciplíny podieľať na výučbe webu [3]. Z tejto iniciatívy sa neskôr stala skupina výskumníkov združených pod názvom Web Science

Trust[1].

Mentormi tejto iniciatívy sú:

- Sir Nigel Shadbolt,
- Sir Tim Berners-Lee,
- Wendy Hallová,
- James Hendler,
- Daniel J Weizner.

Daniel J. Weitzner

1957

Na univerzite MIT sa zaoberá témou decentralizácie údajov a otvorených údajov. Je členom skupiny W3C.

Iniciatíva WSRI sa najviac sústredila na:

1. formulovanie výskumného programu pre širšiu vedeckú komunitu,
2. koordinovanie vývoja vzdelávacích materiálov a študijných programov,
3. zapájanie sa do vedenia tohto rýchlo sa vyvíjajúceho vedného oboru.

Hlavným krédom tejto skupiny sa stalo: Web treba študovať a pochopiť a pritom ho treba vyvíjať. (angl. *Web needs to be studied and understood, and it needs to be engineered*).

Dôvody, prečo by mala vzniknúť táto vedná disciplína a oblasti, ktorým by sa mala venovať, publikovali vo svojom článku v roku 2008 [4]. Tu hovoria o webovede ako novej vednej disciplíne, ktorá ma široký záber a využíva poznatky z mnohých iných vedných disciplín. Taktiež tu hovoria o aktuálnych problémoch webu, a teda najmä o jeho momentálnej zložitosti a neštruktúrovanom charaktere. Tu sa dotýkajú sémantického webu, ako ďalšieho nástupcu, ktorý by mohol priniesť do dnešných neštruktúrovaných informácií nový poriadok.

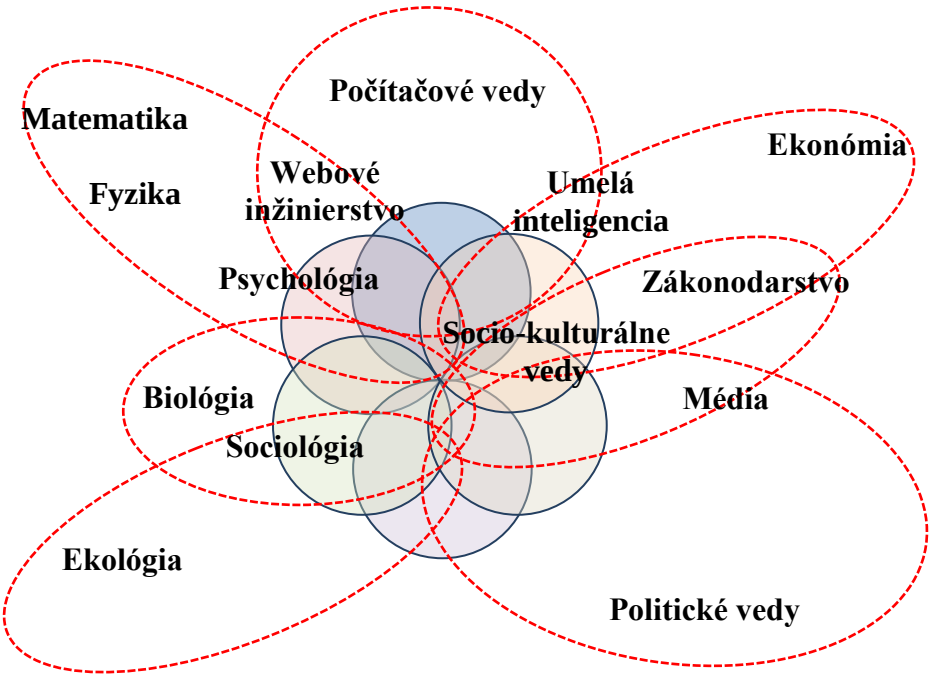
2.2 Záber webovedy

Weboveda ako nová vedná disciplína má veľký prekryv s ostatnými vednými disciplínami. Vo veľkej miere využíva základné poznatky matematiky a fyziky. Takisto využíva aj poznatky z ďalších, zdanlivo nesúvisiacich, vedných disciplín, ako napríklad ekonómia, sociológia, biológia, psychológia a mnohé ďalšie. Prekryv týchto vedných disciplín znázornil *Nigel Shadbolt* na obrázku *1*.

Táto vedná disciplína sa teda venuje problémom dnešného webu. Tieto by sa dali rozdeliť do týchto kategórií:

- štruktúra webu,
- hľadanie významu neštruktúrovaných údajov,
- využívanie potenciálu používateľov webu,
- pravdivosť webu,
- súkromie na webe,
- web a jeho dopad ma myslenie ľudí,
- skrytý web,
- autorské práva na webe,
- sociálne formovanie webu.

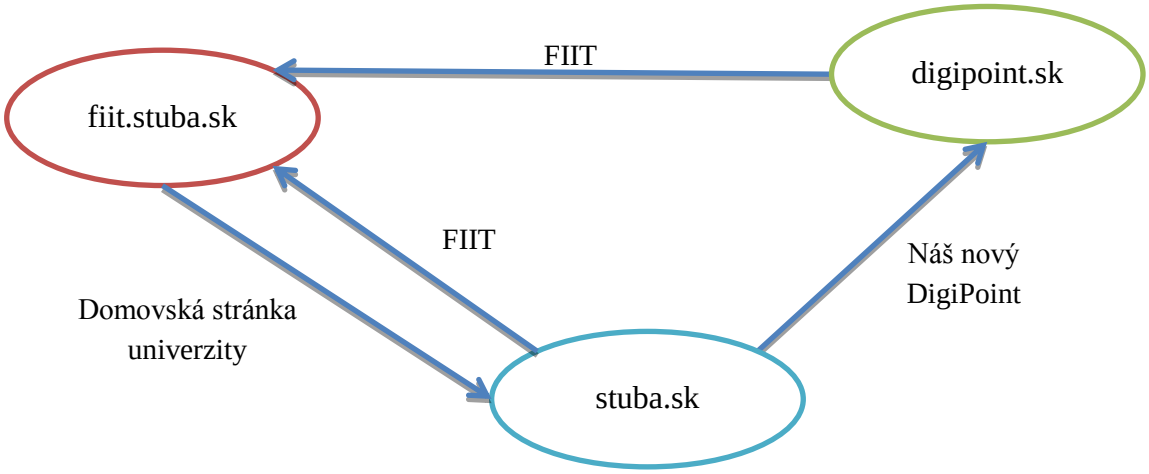
[1] http://webscience.org/



Obrázok 1. Ilustrácia komponentov webovedy, ako medzi-disciplinárneho výskumného poľa. Preložené pôvodné anglické znenie pochádza od Nigel Shadbolta.[2].

2.2.1 Štruktúra webu

Štruktúra dnešného webu sa veľmi podobá grafovej. V štruktúre webu totiž vystupujú webové stránky ako uzly a prepojenia medzi nimi ako hrany, čo je znázornené na obrázku 2. Webový graf má obrovské rozmery a okrem toho rozloženie hrán nemá normálne rozdelenie.

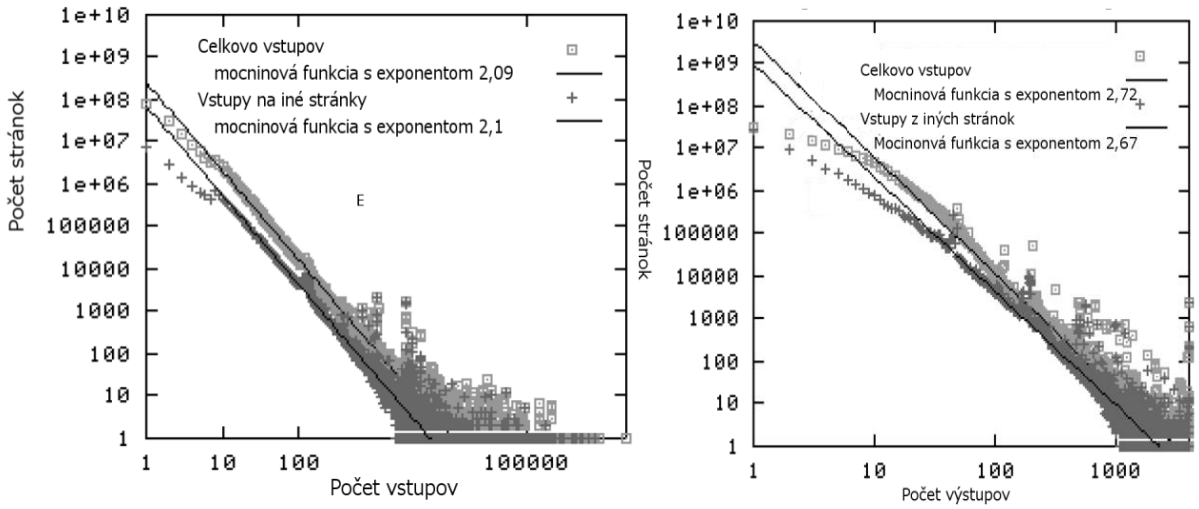


Obrázok 2. Grafová štruktúra webu, v ktorej vystupujú stránky ako uzly a prepojenia medzi uzlami ako linky.

[2] http://dtc.webscience.ecs.soton.ac.uk

Web obsahuje malé množstvo stránok, ktoré majú obrovské množstvo prepojení. Na rozdiel od toho stránok s malým množstvom prepojení je oveľa viacej. Väčšina stránok teda ukazuje iba na malé množstvo dokumentov alebo dokonca neukazujú na žiadne stránky.

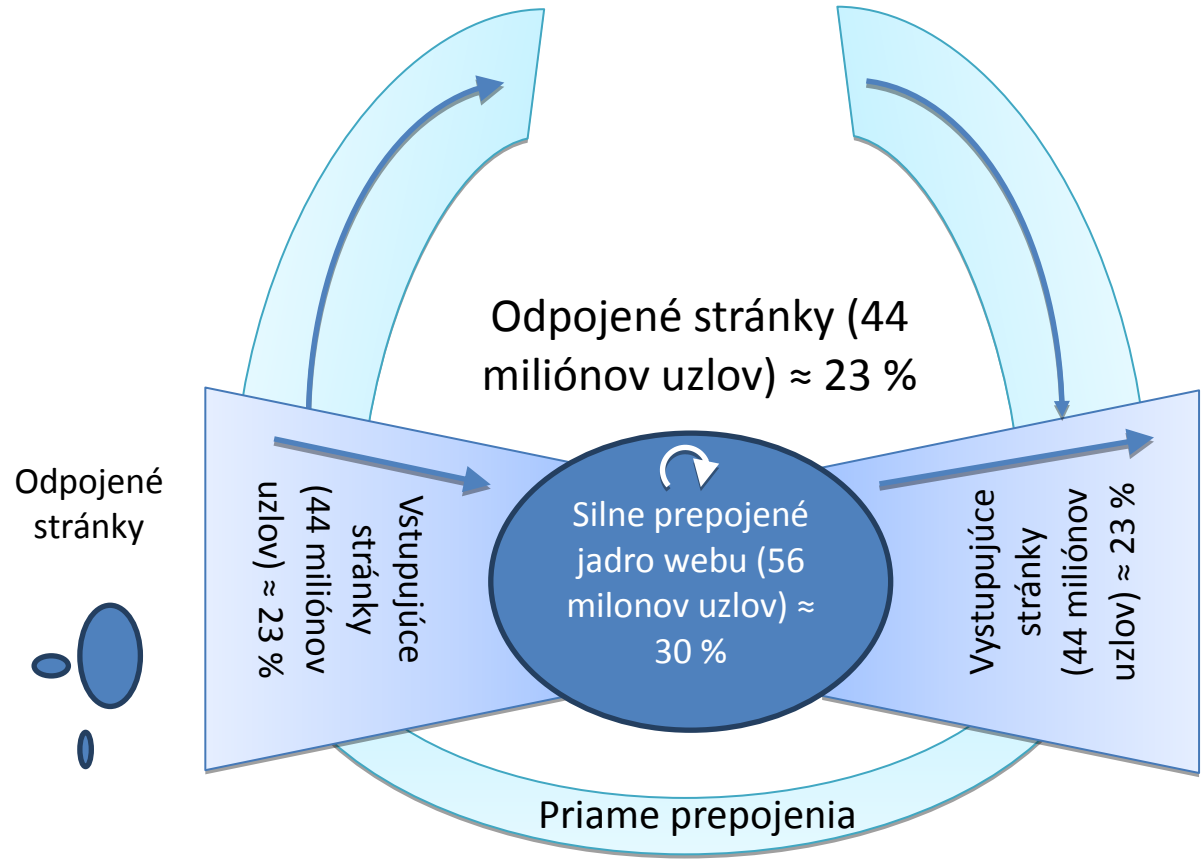
Tento jav preukázali aj pomocou analýzy webového priestoru [4]. Jej výsledky sú znázornené na obrázku 3. Vľavo je pomer počtu stránok k počtu na nich odkazujúcich stránok. Vpravo zase pomer počtu stránok k počtu odkazov na nich umiestnených.



Obrázok 3. Vľavo: graf závislosti počtu stránok a vstupných odkazov, ktoré odkazujú na tieto stránky. Bledošedou sú znázornené výsledky pre všetky odkazy a v tmavošedom zobrazení sú odfiltrované odkazy, ktoré odkazujú na rovnakú doménu. Vpravo: Graf závislosti počtu stránok a výstupných odkazov z týchto stránok. Bledošedou sú znázornené výsledky pre všetky odkazy a v tmavošedom zobrazení sú odfiltrované odkazy, ktoré odkazujú na rovnakú doménu [1].

V tejto štúdii okrem iného opísali tzv. motýlikové rozdelenie webu. Označenie pochádza z faktu, že toto rozdelenie graficky znázornili v podobe motýlika do pánskeho obleku, ako je vidieť na obrázku 4. Tento motýlik vznikol analýzou webových stránok. Od roku 2000, keď poznatky publikovali, analýzu niekoľkokrát zopakovali, pričom percentuálne podiely jednotlivých zložiek sa líšili iba minimálne a narastal iba počet stránok. Aj napriek tomu ide v tomto prípade iba o náčrt toho, ako by takáto štruktúra mohla vyzerať, keďže sa na webe nachádza veľká skrytá časť, ktorú nie je možné preskúmať.

Podľa tejto štúdie je jadrom webu silne prepojené jadro stránok. Tieto stránky sú navzájom dobre poprepájané a navzájom na seba odkazujú väčším množstvom odkazov. Na ľavej stránke motýlika sa potom nachádzajú vstupné stránky. Tieto stránky ukazujú na stránky v silne prepojenom jadre, ale na ne v tomto jadre neukazuje žiadna stránka. Okrem toho tieto stránky odkazujú na ďalšie stránky, ktoré však už neukazujú na žiadne stránky a sú znázornené na obrázku vľavo hore.



Obrázok 4. Rozdelenie stránok na webe. Stred motýlika tvoria silno prepojené stránky. Majú veľa odkazov a zároveň odkazujú na veľké množstvo stránok. Na stranách sú vstupujúce stránky a vystupujúce stránky. Stránky zo silno prepojeného jadra na vstupujúce stránky neukazujú a vystupujúce stránky zase neukazujú na stránky v silne prepojenom jadre. Na spodku motýlika sú prepojenia medzi vstupujúcimi a vystupujúcimi stránkami. V hornej časti obrázku sú stránky ktoré buď na nič neukazujú, ale na ne ukazujú vstupujúce stránky alebo ukazujú len na vystupujúce stránky a na ne neukazuje nikto. Okrem toho sa v obrázku nachádzajú aj "pečeňové škvrny" v podobe odpojených sústav stránok [4].

Na pravej strane sa potom nachádzajú stránky, na ktoré ukazujú stránky v silne prepojenom jadre ale tieto stránky ďalej na nič neukazujú. Na tieto stránky však odkazuje niekoľko ďalších stránok (vpravo hore). Na tieto stránky (vpravo hore) však už neodkazuje nikto ďalší. Okrem toho sa v spodnej časti motýlika nachádzajú aj prepojenia medzi vstupujúcimi a vystupujúcimi stránkami.

Poslednú množinu tvoria stránky, ktoré sú takpovediac odpojené od veľkého webu. Tieto stránky tvoria menšie podgrafy prepojení, nepripájajú sa však k celkovému grafu webu.

2.2.2 Skrytý web

Pri bežnej práci s webom sa dostávame iba k zdrojom, ktoré sa dnešným vyhľadávačom podarí zaindexovať. Pred týmito vyhľadávačmi je však veľké množstvo stránok skrytých. Na tieto stránky nikto neukazuje a preto ich nie je možné klasickými prístupmi nájsť. Niektoré zdroje odhadujú, že skrytý web obsahuje 96% celkovej veľkosti webu a pre bežných používateľov sú dnes viditeľné a dohľadateľné iba 4% [6]. Iné štatistiky udávajú že viditeľná je 1/3 webu a 2/3 sú pred nami skryté [7]. Práve preto sa pomer skrytých a verejných častí webu často prirovnáva k ľadovcu, ako na obrázku 5. Ľadovec totižto skrýva veľkú časť svojho obsahu pod hladinou mora, ktorú nie je možné vidieť. Toto sa dá dobre prirovnať k skrytému webu, ktorý je tiež ukrytý pod hladinou.

Obrázok 5. Ľadovec ako prirovnanie k obsahu webu. Nad hladinou sa nachádza verejne prístupná časť, pod hladinou je rozľahlá časť skrytého webu.

Medzi skrytý obsah patria napríklad súkromné časti webových stránok. Nimi sú napríklad obsahy diskusných fór, spoplatnené časti stránok alebo len časti stránok s nutnou registráciou. Niektoré z týchto stránok už takéto problémy riešia pomocou udeľovania prístupov pre vyhľadávače, ktoré tak dokážu zindexovať ich obsah a pri vyhľadávaní ukážu časť stránky používateľovi. Medzi skryté stránky patria aj veľké štátne databázy, ktoré sa síce na webe nachádzajú, ale často sú neprístupné pre bežných používateľov. V dnešnej dobe sa však aj tieto databázy dostávajú bližšie k ľuďom a štáty začínajú zverejňovať takéto informácie.

Okrem týchto informácií sa v skrytom webe ukrývajú rôzne vygenerované dopyty - napríklad v internetových obchodoch. Používatelia týchto obchodov vytvárajú dopyty do databázy pomocou vyberania parametrov. Tým vždy vytvárajú novú a novú stránku s čiastočne pozmeneným obsahom. Časťou skrytého webu sú aj rôzne dokumenty, ktoré si medzi sebou posielajú používatelia pomocou rôznych FTP účtov alebo služieb na zdieľanie údajov.

Ďalej do tejto skrytej časti webu môžeme zaradiť dnes populárne siete s nelegálnym obsahom, ako sú P2P siete, warez a podobne. Veľkú časť týchto údajov tvorí aj obsah siete Onion. Onion (slov. cibuľa) je sieť, ktorá zabezpečuje vysokú anonymitu svojich používateľov. Po pripojení do siete sa používateľova identita skrýva za niekoľko prístupových bodov. Jeho komunikácia s cieľovou destináciou sa niekoľkonásobne zabaľuje do ďalšej a ďalšej komunikácie, čo vytvára cibuľovitý efekt.

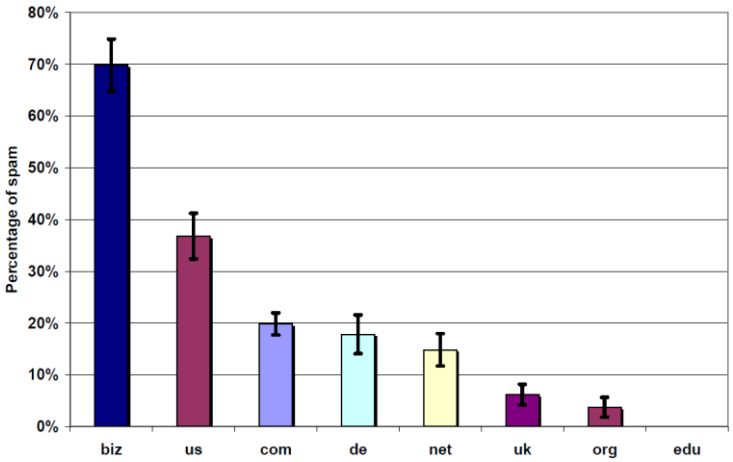
Totožnosť takéhoto používateľa sa dá potom ťažko vypátrať. Práve preto sa tento druh sietí často využíva na nelegálne činnosti. Vyhľadávače sa do takejto siete nevedia dostať a takisto by boli zindexované výsledky z tejto siete pre bežných používateľov nepoužiteľné, keďže by sa do siete nevedeli pripojiť.

2.2.3 Redundancia na webe

Web ako enormne veľká databáza obsahuje aj veľké množstvo rovnakých alebo veľmi podobných stránok. Tieto stránky sú rovnaké buď z podstaty kopírovania rovnakých textov v prípade spravodajských serverov alebo umiestňovaním rovnakých dokumentov na viaceré úložiská.

Výskumy [8] preukázali, že 30 % stránok je navzájom rovnakých alebo veľmi podobných. Pri tomto výskume stiahli 150 miliónov webových stránok. Toto sťahovanie opakovali 11 týždňov. Následne stránky navzájom porovnávali.

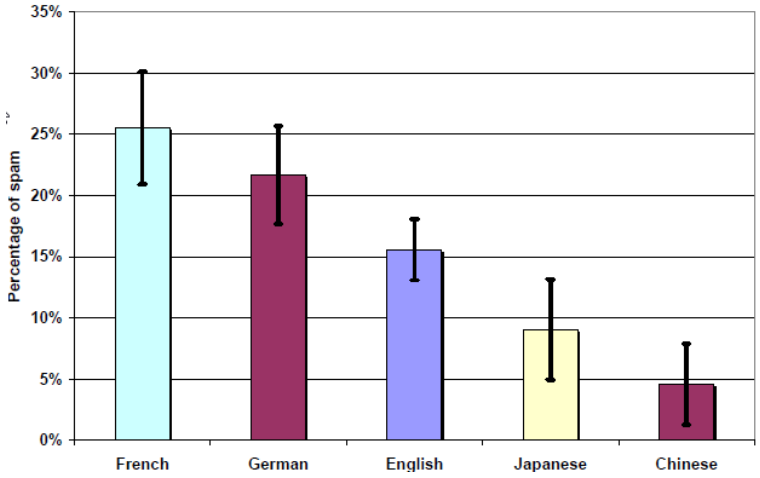
Ďalší zaujímavý výskum [9] sa zameral na zisťovanie podielu stránok, ktoré obsahujú iba nevyžiadaný reklamný obsah (angl. spam). Tento výskum uskutočnili na 105 miliónoch stránok stiahnutých pomocou preliezača Bing. Pri výskume využili heuristiky tohto vyhľadávača na odhaľovanie stránok obsahujúcich spam. Podľa výskumov potom nevyžiadanú reklamu celkovo obsahuje asi 14% stránok. Výskum sa zaoberal aj rozdelením reklamy podľa toho, z akej domény je daná stránka (Obrázok 6) alebo podľa jazyka (Obrázok 7), v ktorom je napísaná.



Obrázok 6. Percentuálny obsah nevyžiadanej reklamy na stránkach s rozdelením podľa domény.

Najviac tejto reklamy sa nachádza na stránkach z domény biz, ktorá sa predvolene používa na stránky týkajúce sa podnikania. V prípade domény biz to bolo až 70% z celkového množstva stránok s touto doménou. Nasledovala doména us, kde to už bolo len okolo 36% a následne do-

ména com, v ktorej prípade to bolo už len 20%. Najmenej reklamy sa podľa tohto výskumu nachádzalo na stránkach s doménou uk a org. Pričom takmer žiadna reklama nebola na stránkach z domény edu, ktorá je rezervovaná pre vzdelávacie účely.



Obrázok 7. Percentuálny obsah nevyžiadanej reklamy na stránkach s rozdelením podľa jazyka použitého na stránke.

V prípade jazykov je najviac reklamy na stránkach písaných po francúzsky a nemecky. Zaujímavé je, že veľmi používaný jazyk angličtinu zasiahla reklama iba v 16%. Najmenej reklamy sa ale nachádza na japonských (menej ako 9%) a čínskych (menej ako 5%) stránkach.

2.3 Zhrnutie

V tejto kapitole sme ukázali dnes veľmi dobre rozvinutý potenciál webu. Práve preto si jav ako je web zaslúži novú vednú disciplínu, ktorá sa bude venovať jeho výskumu. Takisto sme ukázali, kde začal web samotný ale aj základy vedy, ktorá sa jeho výskumom venuje. Rozobrali sme aj zložitú štruktúru webu a jednotlivé časti, ako veľmi prepojené stránky, vstupné stránky, výstupné stránky, ale aj odpojené stránky. Venovali sme sa aj prvkom, ako je skrytý web a dnes veľmi častá reklama na webe. V nasledujúcich kapitolách sa budeme venovať ďalším zaujímavým prvkom, ktoré sa na dnešnom webe využívajú a ktorým web vďačí za jeho úspech.

Literatúra

[1] Temple, K.: What Happens in a Internet Minute? (2012). Dostupné na: http://scoop.intel.com/what-happens-in-an-internet-minute/.

[2] How many addresses can IPv6 hold? (2012). Dostupné na: http://itsnobody.wordpress.com/2012/02/17/how-many-addresses-can-ipv6-hold/.

[3] About Us | Web Science Trust. (2013). Dostupné na: http://wstweb1.ecs.soton.ac.uk/web-science-trust/about-us/.

[4] Hendler, J., Shadbolt, N., Hall, W., Berners-Lee, T. and Weitzner, D.: Web science: an interdisciplinary approach to understanding the web. In *Communications of ACM*, vol. 51, no. 7, (2008), pp. 60-69.

[5] Kumar, , et al. The Web as a graph. In *Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS '00)*, (2000), pp. 309–20

[6] Bigney, T.: Traversing the deep web. (2012). Dostupné na: http://www.tylerbigney.com/p/blog.html

[7] He, B. et al. Accessing the deep web. *In Communications of the ACM - ACM at sixty: a look back in time*, vol. 50, issue 5, (2007), pp. 97-101.

[8] Fetterly et al.: On the evolution of clusters of near-duplicate web pages. In *Journal of Web Engineering*, vol. 2, no.4, (2004), pp. 228-246.

[9] Ntoulas et al.: Detecting spam web pages through content analysis, In *Proceedings of international conference on World Wide Web 2006*, (2006).

3 Architektúra webu

Web je obrovskou sieťou zdrojov navzájom prepojených takzvanými odkazmi. Na sprístupňovanie týchto zdrojov používame pomerne jednoduché nástroje a protokoly. Zabezpečujú prístup k dokumentom, ktorý je efektívny, škálovateľný a odolný voči chybám. V tejto kapitole sa budeme venovať tomu, čo je to zdroj na webe, ako sa identifikuje, ako k nemu môžeme pristúpiť a čo presne sa deje pri pristupovaní k zdroju na webe. Pozrieme sa aj na obmedzenia súčasných prostriedkov a na smerovanie ich vývoja v najbližšej budúcnosti.

Jednou z najčastejších chýb, s ktorou sa stretávame, ak niekto rozpráva o webe je to, že voľne zamieňa termín web a internet, pričom ich význam je diametrálne odlišný. World Wide Web (WWW) alebo skrátene web sa dá definovať ako informačný priestor, v ktorom sú jednotlivé zdroje identifikované jedinečným identifikátorom URI a sú navzájom prepojené prostredníctvom odkazov.

Naproti tomu internet, tak ako sa definuje v Oxfordskom slovníku [1], je celosvetová sieť počítačov, ktorá poskytuje rad informačných a komunikačných prostriedkov, zložená z prepojených sietí pomocou štandardizovaných komunikačných protokolov. Web je teda len jedna zo služieb, ktoré fungujú v prostredí internetu a predstavuje sieť prepojených zdrojov v podobe dokumentov alebo služieb.

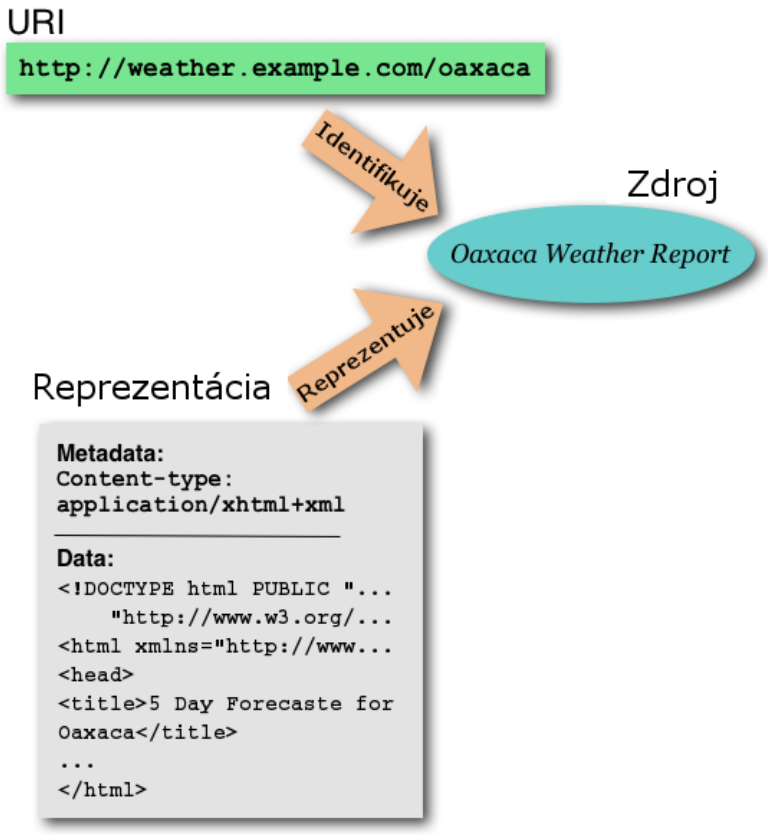
3.1 Základné kamene webu

Pri sprístupňovaní údajov na webe sa používa niekoľko základných princípov, pomocou ktorých je možne pristúpiť k jednotlivým zdrojom. Základom pre pristúpenie k zdrojom je ich identifikácia a interakcia medzi agentom, ktorý chce pristúpiť k zdroju a službou, ktorá tento zdroj poskytuje.

3.1.1 Identifikácia a interakcia s webovými zdrojmi

Jedným zo základných cieľov pri vzniku webu bolo vytvoriť priestor, kde by mohol hocikto zdieľať informácie s hocikým iným. Na dosiahnutie tohto cieľa sa používa globálny identifikačný systém - URI, pomocou ktorého môže každý jedinečne označiť svoj zdroj tak, aby naň mohli ostatní používatelia odkázať. Aby sa zabránilo možnému vzniku problémov spojených s kolíziami URI identifikátorov, zaviedli prideľovanie URI a vlastníctvo URI, vďaka ktorým majú len vlastníci identifikátorov právo použiť tieto identifikátory na označenie svojich zdrojov.

Vo všeobecnosti platí, že každý zdroj sa identifikuje jedinečným URI, pričom nie je presne špecifikované, čo to zdroj presne je. Jedinou podmienkou pre zdroj je, že jeho reprezentáciu musí byť možné zaslať v podobe správ. Je dôležité uvedomiť si rozdiel medzi zdrojom a jeho reprezentáciou. V prípade, ak agent (ľudia alebo softvér pohybujúci sa v informačnom priestore webu) pristúpi k zdroju, získa len jednu jeho reprezentáciu, pričom jeden zdroj môže mať viacero rôznych reprezentácií. Takýmito reprezentáciami môžu byť napríklad rôzne formáty jedného zdroja. Situáciu schematicky znázorníme na obrázku 8.



Obrázok 8. Vzťah medzi zdrojom, jeho identifikáciou pomocou URI a reprezentáciou. Prevzaté a preložené z [2] Architecture of the World Wide Web (Second Edition).

Pri pristupovaní k zdrojom cez sieť používajú agenty štandardizované protokoly (napr. HTTP, FTP, SOAP, SMTP), ktoré umožňujú pristúpiť k reprezentácii špecifikovaného zdroja. Repre-

zentácia zdroja je sprístupnená pomocou správ, ktoré obsahujú údaje, ako aj k nim pripojené opisné údaje, ako je napríklad použité kódovanie, formát alebo jazyková mutácia.

3.1.2 Vzťah URI k URL a URN

URI sa používa ako identifikátor zdrojov na webe. URI môže mať podobu URL alebo URN, pričom URN identifikuje zdroj na základe jeho názvu v špecifikovanom mennom priestore. Napríklad URN `urn:isbn:0-395-36341-1` identifikuje knihu na základe medzinárodného identifikátora ISBN, ale nijako nenaznačuje, ako a kde je možné nájsť kópiu tejto knihy. Naopak URL okrem identifikácie zdroja poskytuje aj základné informácie potrebné na sprístupnenie tohto zdroja. Napríklad URL `http://example.org/wiki/Main_Page` odkazuje na zdroj identifikovaný ako `wiki/Main_Page`, ktorý je prístupný pomocou HTTP protokolu na počítači v sieti, ktorého doménové meno je `example.org`.

3.2 Čo sa deje, keď nasledujem odkaz?

Základnou operáciou pri pohybovaní sa vo webovom priestore je presúvanie sa medzi jednotlivými webovými stránkami zobrazenými v prehliadači prostredníctvom nasledovania odkazov. Pri zobrazení jednej webovej stránky však prehliadač vykonáva sériu úloh spojených so získaným obsahu stránky, jej ďalších súčastí a ich zobrazením. Postupnosť krokov, ktoré sa dejú pri zobrazovaní webového zdroja v prehliadači opíšeme v tejto časti na príklade zobrazenia bežnej webovej stránky.

Zadanie URL adresy

Prvý krok je pomerne jednoduchý a každému známy: zadanie URL adresy do prehliadača. Pre potreby tohto príkladu môžeme povedať, že touto adresou bude adresa dokumentu, ktorý opisuje pravidlá pre prijímanie na štúdium na Fakulte informatiky a informačných technológií STU v Bratislave `http://www.fiit.stuba.sk/generate_page.php?page_id=353`

Vyhľadanie IP adresy pre doménové meno

V ďalšom kroku je potrebné identifikovať adresu počítača v sieti, ktorý je schopný poskytnúť nám zdroj, ktorý hľadáme a ktorý je identifikovaný pomocou URL, ktorú sme zadali do prehliadača. Na základe doménového mena získaného z URL adresy sa snažíme nájsť DNS (Domain Name System) záznam postupne v pamäti prehliadača, pamäti operačného systému, pamäti smerovača, prostredníctvom ktorého je počítač pripojený do internetu. Ak sa daný záznam nenájde, tak nasleduje rekurzívne vyhľadávanie v DNS serveroch.

Odoslanie HTTP dopytu na server

Pre potreby tohto príkladu môžeme predpokladať, že hľadanú stránku sme nikdy nezobrazovali a teda sa nenachádza v žiadnej dočasnej pamäti prehliadača. V URL adrese, ktorú sme napísali

do prehliadača sme identifikovali, že hľadaný zdroj je dostupný prostredníctvom HTTP protokolu. Prehliadač teda odošle požiadavku na získanie tohto zdroja na server. Pri odosielaní požiadavky použije HTTP GET metódu, identifikuje sa reťazcom špecifickým pre prehliadač, ktorý používame, odošle informácie o formátoch odpovedí, ktoré akceptuje a nechá otvorené TCP spojenie pre ďalšiu komunikáciu so serverom.

Server spracuje požiadavku

Server prijme HTTP GET požiadavku na konkrétny zdroj, spracuje ju a odošle odpoveď. Toto je na prvý pohľad pomerne jednoduchá a priamočiara úloha, ale v skutočnosti je to komplikovaný proces napríklad kvôli používaniu vyrovnávacích pamätí a dynamickému generovaniu obsahu.

Server odošle odpoveď

Server spracoval požiadavku a odoslal späť odpoveď. V tele odpovede je samotný HTML dokument a v hlavičke sa nachádzajú ďalšie metaúdaje, ako je napríklad HTTP Status Code 200, ktorý hovorí o tom, že celé spracovanie prebehlo v poriadku, informácie o kódovaní a formáte odoslaných údajov.

Prehliadač vykresľuje dokument

Po tom, ako prehliadač získal HTML dokument, začal ho vykresľovať. V tomto kroku by sa základný cyklus spracovania požiadavky skončil, ale keďže pri spracovávaní požiadaviek na webové stránky je bežné, že tieto odkazujú na ďalšie zdroje potrebné pre správne vykreslenie stránky, v tomto príklade opíšeme aj tie.

Prehliadač odoslal požiadavky na prvky obsiahnuté v dokumente

Pri spracovávaní a zobrazovaní HTML stránky prehliadač narazil na rôzne elementy ako sú napríklad obrázky CSS súbory alebo javascriptové súbory, ktoré sú potrebné pre správny výzor alebo fungovanie stránky. Každý z týchto súborov je identifikovaný pomocou URL a prehliadač ho získava podobným procesom, ako to bolo pri samotnej HTML stránke.

Prehliadač posiela asynchrónne požiadavky

Veľké množstvo moderných stránok používa javascript na to, aby zobrazovali dynamický obsah. Na zobrazenie tohto obsahu musí prehliadač častokrát komunikovať so serverom aj po tom, ako vykreslil celú stránku. Takéto dodatočné požiadavky vyvolané až po vykreslení stránky sa nazývajú skratkou AJAX (Asynchronous JavaScript And XML).

3.3 Vývoj „webových“ protokolov

Základným kameňom webu tak, ako ho teraz poznáme je HTTP [3] protokol. HTTP je v súčasnosti asi najznámejším internetovým protokolom. Umožnil pomerne jednoduchý prístup

k webovým stránkam a ďalším zdrojom na webe. Prvá verzia (verzia 0.9) tohto protokolu vznikla v roku 1991. Táto verzia poskytovala len najzákladnejšie funkcie: umožňoval na základe URL adresy získať odpoveď. V roku 1996 vznikla verzia 1.0, ktorá tento protokol obohatila o množstvo funkcií. Jednou z najdôležitejších bola hlavička, ktorá poskytuje informácie o každej správe, ktorá sa pomocou tohto protokolu prenáša. To umožnilo posielať pomocou protokolu ďalšie formáty údajov, ako sú napríklad obrázky a podobne. V tejto verzii pribudli tiež stavové kódy, ktoré umožnili informovať o rôznych stavoch, ktoré nastali pri spracovávaní požiadavky ako napríklad kód 200, ktorý označuje bezchybne vykonaný dopyt, kódy skupiny 3xx, ktoré označujú rôzne typy presmerovania, skupiny 4xx, ktoré označujú problémy s prístupom alebo kódy skupiny 5xx, ktoré označujú problémy na strane servera.

Verzia 1.1 z roku 1999 sa stala štandardom, ktorý sa používa dodnes a pridala do protokolu ďalšie populárne vlastnosti, ako je napríklad kompresia odpovedí.

3.3.1 Moderné webové protokoly

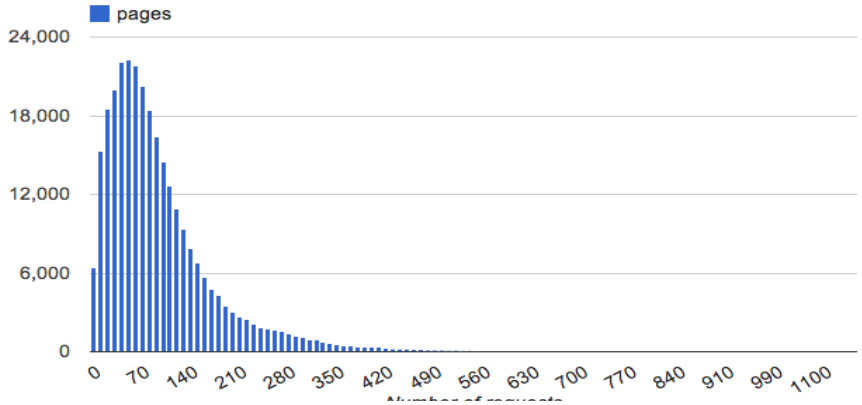
Od roku 1999, keď vznikla aktuálne používaná verzia HTTP protokolu, sa pri vytváraní webového obsahu a pristupovaní k nemu veľa zmenilo. S nástupom takzvaného Webu 2.0 sa vo veľkej miere začala používať technológia AJAX, ktorá síce nie je súčasťou HTTP protokolu, ale významne zmenila spôsob, ako pracujeme s webovými stránkami. S použitím AJAXu je možné dynamicky meniť obsah stránky po tom, ako bola po prvýkrát vykreslená, čo umožnilo vznik množstva zaujímavých aplikácií a služieb.

V súčasnosti sa do popredia dostáva množstvo ďalších protokolov, ktoré poskytujú ďalšie možnosti pre zobrazovanie obsahu a pre interakciu s obsahom. V tejto časti opisujeme len krátky výber najpoužívanejších z týchto protokolov:

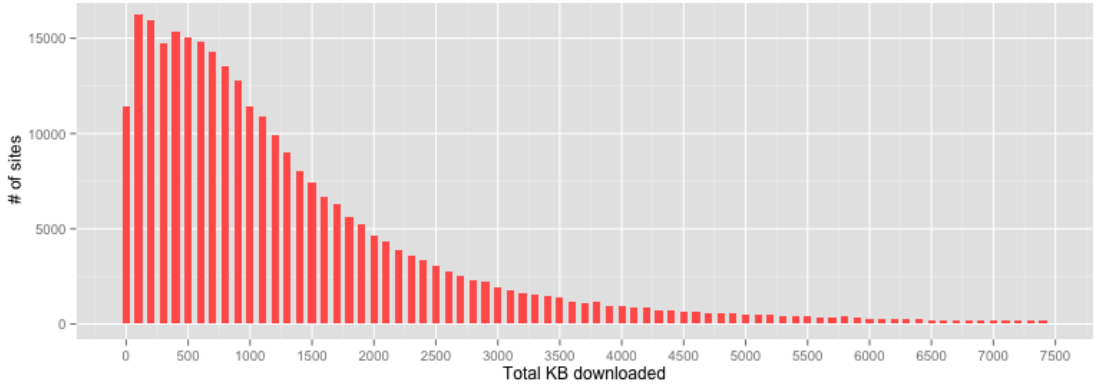
- **WebSockets** protokol napríklad umožňuje obojsmernú komunikáciu medzi klientom a serverom v reálnom čase. Komunikácia sa zabezpečuje prostredníctvom stále otvoreného spojenia, čo výrazne znižuje časy potrebné na doručenie jednotlivých správ. Tento protokol si našiel uplatnenie v rôznych aplikáciách, ktoré vyžadujú rýchlu komunikáciu, od rôznych služieb na posielanie správ cez služby na online editovanie dokumentov až po rôzne hry.
- **WebGL** je protokol, ktorý umožňuje v prehliadači zobrazovať zložité grafické objekty, pričom tieto sa vytvárajú na strane klienta s použitím jeho grafickej karty. Tento protokol sa používa v rôznych grafických aplikáciách ale napríklad aj pre vykresľovanie máp.
- **WebRTC** je protokol a rozhranie pre jazyk javascript, ktoré umožňujú tvorbu aplikácií, ktoré využívajú komunikáciu (audio, video) v reálnom čase. Pomocou tohto rozhrania je možné pomerne jednoducho vytvoriť aplikáciu, ktorá využíva audiovizuálnu komunikáciu medzi jej používateľmi.

3.3.2 HTTP 2.0

Od roku 1999 sa forma webových stránok výrazne posunula a už to nie sú len jednoduché HTML dokumenty, ku ktorým je pripojených zopár obrázkov a CSS štýlov. Veľká časť spracovania sa posunula na stranu klienta a výrazne sa zväčšila celková veľkosť stránok spolu s počtom ich súčastí. Podľa HTTP Archive crawl data [3] dnes priemerná webová stránka potrebuje na správne vykreslenie 76 dopytov na rôzne zdroje ako sú obrázky, CSS súbory a podobne, pričom prenesie takmer 1 megabajt údajov, ktoré získa z jedenástich rôznych domén. Toto je veľmi veľa údajov najmä pri pristupovaní k stránkam z rôznych mobilných zariadení, ktoré majú stále ešte relatívne pomalé a drahé pripojenia. Navyše spomalenie zobrazovania stránok spôsobuje aj obmedzenie paralelného sťahovania zdrojov, ktoré je vo všetkých súčasných prehliadačoch obmedzené na 6 paralelných spojení. Presnejšiu predstavu o počte zdrojov potrebných na vykreslenie jednej stránky je možné si vytvoriť na základe grafu na obrázku 9 a predstavu o množstve údajov potrebných pre vykreslenie jednej stránky je možné si vytvoriť na základe diagramu na obrázku 10.



Obrázok 9. Histogram počtu stránok podľa počtu rôznych zdrojov potrebných na ich vykreslenie. Prebraté z HTTP Archive crawl data [4].

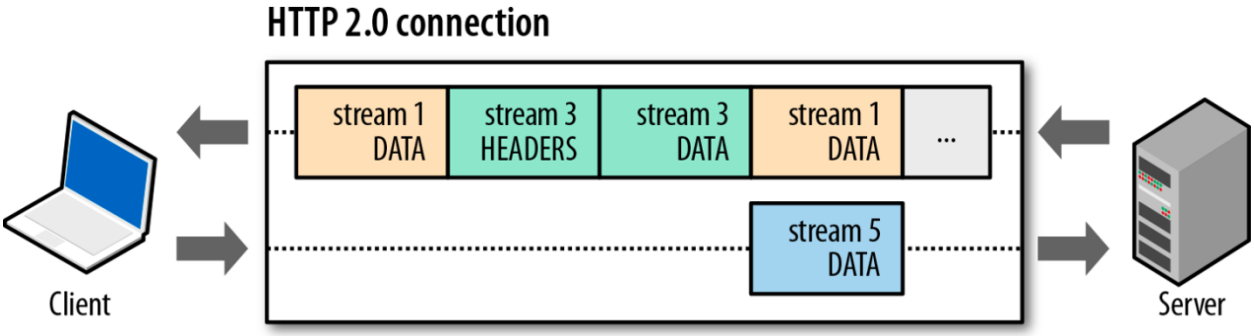


Obrázok 10. Histogram počtu stránok podľa objemu údajov potrebných na ich vykreslenie. Prebraté z HTTP Archive crawl data [4].

Na znižovanie objemu a zmenšovanie počtu rôznych zdrojov potrebných na vykreslenie jednotlivých stránok sa v súčasnosti používa viacero techník ako napríklad spájanie CSS súborov, spá-

janie javascriptových súborov, spájanie obrázkov alebo vkladanie dodatočného obsahu priamo do tela stránky. Tieto metódy však vždy majú negatívny dopad na náročnosť spracovania stránky na strane servera alebo klienta a v neposlednom rade na náročnosť implementácie takýchto stránok. Jedno z možných riešení by mohla byť pripravovaná nová verzia HTTP protokolu, HTTP 2.0.

Najvýznamnejšou zmenou, ktorú tento protokol zavádza, je využívanie jedného spojenia na posielanie všetkých zdrojov potrebných na vykreslenie stránky z jednej domény. Ak doteraz bolo potrebné získať napríklad 97 zdrojov z 11 rôznych domén, pomocou tohto protokolu bude potrebné vytvoriť len 11 spojení (pre každú z domén) a prostredníctvom nich odoslať všetky potrebné údaje. Ďalšie funkcie, ktoré tento protokol prináša, je prioritizácia zdrojov, kompresia hlavičiek alebo kontrola prúdu údajov. Schematický náčrt spojenia pomocou HTTP 2.0 protokolu je zobrazený na obrázku číslo 11.



Obrázok 11. Schematický náčrt spojenia medzi klientom a serverom pomocou HTTP 2.0 protokolu. Prebraté z High Performance Browser Networking [5].

Nová verzia HTTP protokolu je zatiaľ len návrhom, niektoré služby ako napríklad twitter ju však už používajú. Rovnako všetky moderné prehliadače už zaviedli podporu pre tento protokol. Táto verzia zavádza priamu podporu pre viacero techník, ktoré doposiaľ museli programátori prácne simulovať. Jeho rozšírenie by sa maloprejaviť zvýšením rýchlosti zobrazovania stránok a znížením objemu prenášaných údajov, čo uvítajú najmä používatelia rôznych mobilných zariadení.

3.4 Zhrnutie

Web mohol dosiahnuť svoju súčasnú veľkosť vďaka jednoduchosti, škálovateľnosti a odolnosti voči chybám. Tieto vlastnosti sa opierajú o základné kamene, na ktorých web stojí. Vďaka jednotnej identifikácií zdrojov pomocou URI je možné identifikovať každý zdroj na webe a prepájať ich medzi sebou pomocou odkazov. Prístup k týmto zdrojom je zabezpečený pomocou rôznych protokolov, z ktorých najznámejší je HTTP. Postupom času sa protokol HTTP vyvíjal spolu s meniacimi sa požiadavkami na sprístupňovanie informácií na webe. V práci sme zhrnuli najdôležitejšie vlastnosti týchto základných kameňov webu (URI a HTTP), ako aj ich postupné rozširovania a načrtli sme aktuálne požiadavky a obmedzenia webu a smerovanie, kam by sa mohol uberať v najbližšom čase.

Literatúra

[1] Stevenson, A.: *Oxford dictionary of English*. Oxford University Press, (2010).

[2] Thompson H.: *Architecture of the World Wide Web (Second Edition)*. (2014). Dostupné na: http://w3ctag.github.io/webarch/

[3] R. Fielding et al: *Hypertext Transfer Protocol - HTTP/1.1*. (1999). Dostupné na: http://www.ietf.org/rfc/rfc2616.txt

[4] HTTP Archive crawl data, (2014). Dostupné na: http://bigqueri.es/t/calculate-medians-for-latest-http-archive-run/7/5

[5] Grigorik, I.: *High Performance Browser Networking*, O'Reilly Media, (2013).

4 Charakteristiky webu

V tejto kapitole predstavujeme charakteristiky webu najmä na príklade dvoch najvýznamnejších štúdií zaoberajúcich sa charakterizáciou webu – W3C Characterization Activity a OCLC Characterization Research. Zaoberáme sa charakteristikami ako sú veľkosť, rast, jazykové a krajinné rozdelenie webových stránok, ale aj ich štruktúra, obsah a správanie sa používateľov na webe. Okrem vtedajších výsledkov uvádzame aj niektoré súčasné hodnoty z monitorovania webu. Ďalej komentujeme dve štúdie zamerané na dynamickú povahu webu. Je to neustále meniace sa prostredie, v ktorom vznikajú, zanikajú a menia obsah webové stránky aj prepojenia medzi nimi. S tým súvisí aj problém mŕtvych prepojení, ktorým sa zaoberá časť výskumníckej komunity. V závere kapitoly spomíname tzv. blogosféru – časť webu, do ktorej sa dnes sústredí veľká časť činnosti používateľov na webe.

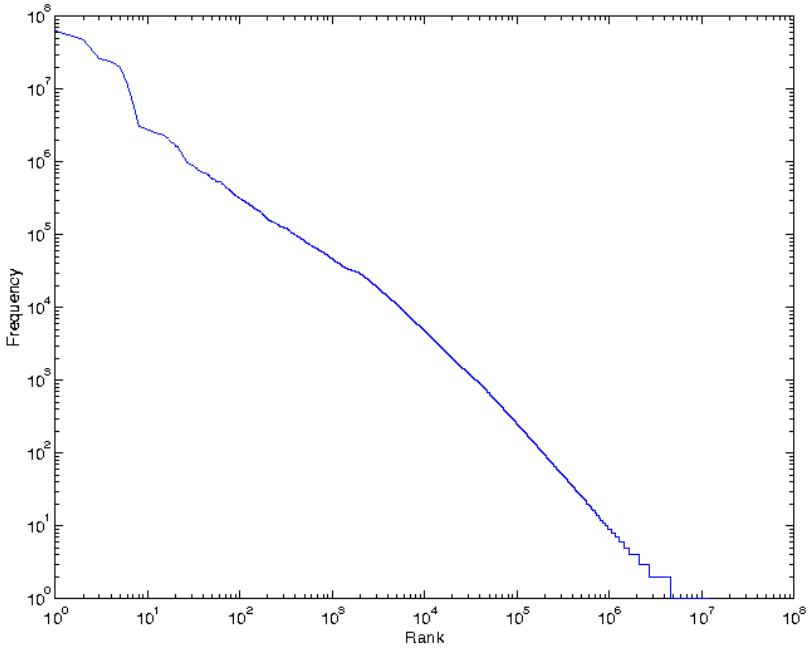
4.1 Činnosť W3C zameraná na charakterizovanie webu

Prvý pokus o charakterizovanie webu bol výsledkom výskumu *Činnosť W3C zameraná na charakterizovanie webu* (angl. *Web Characterization Activity*) [6] z rokov 1998-99 komunity združenej v konzorciu W3C (World Wide Web Consortium). Táto komunita zaviedla prvé jednoznačné definície pojmov webu ako zdroj, linka, proxy, klient, server, správa, požiadavka a pod. Toto jednoznačné vymedzenie pojmov slúžilo na podporu výskumu v oblasti webu a umožnilo vytvárať, interpretovať a porovnávať rôzne webové metriky používané pri opise webu.

Výskum sa okrem zjednotenia terminológie zaoberal charakterizovaním štruktúry webu a správania sa používateľov na webe. V rámci výskumu sledovali činnosť na klientoch, serveroch a proxy serveroch. Výsledná správa [6] uvádza zistenia, že aj dynamické prostredie, akým je web, sa v niektorých ohľadoch správa pravidelne a predvídateľne.

Najvýznamnejším zistením bolo, že popularita stránok sa riadi tzv. Zipfovým rozdelením. V praxi to znamená, že existuje zopár veľmi obľúbených (často navštevovaných) stránok,

stredné množstvo priemerne obľúbených stránok a obrovský počet stránok, ktoré málokedy niekto navštívi. Zipfovo rozdelenie predtým použili napríklad na charakterizovanie distribúcie slov v prirodzenom jazyku, kde sa podobne nachádza pár veľmi často používaných slov, bežná slovná zásoba a obrovské množstvo sporadicky využívaných slov. Iným príkladom použitia je na vyjadrenie popularity kníh v knižnici. Na obrázku 12 je znázornené Zipfovo rozdelenie pre súbor stránok zoradených podľa popularity (počet návštev). Zipfovo rozdelenie platí, aj ak sa popularita stránok nemeria počtom návštev, ale napríklad počtom odkazujúcich stránok ako pri algoritme Page Rank.

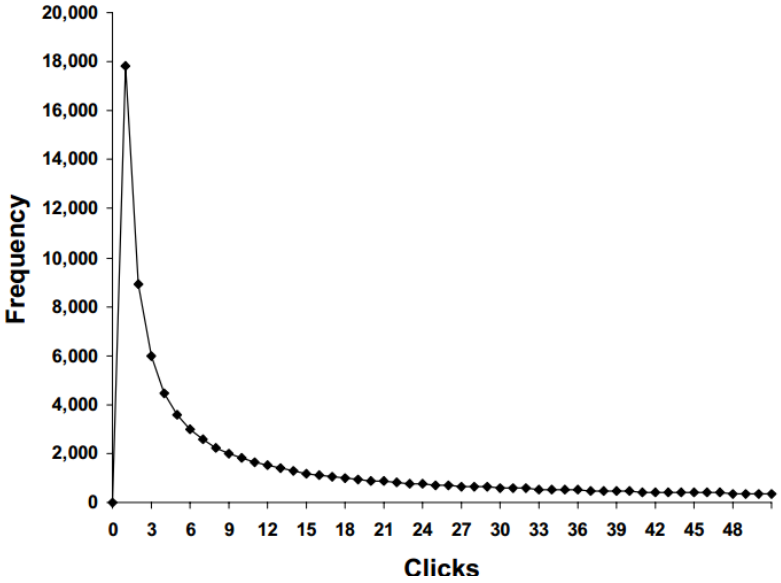


Obrázok 12. Zipfovo rozdelenie stránok AOL podľa počtu návštev za päť dní v decembri 1997 (log-log škála) [6].

Ďalšími objavenými vlastnosťami webu, ktoré sa riadia pevným rozdelením, boli čas prenosu stránky zo servera na klienta (Paretovo rozdelenie), veľkosť stránok (Paretovo rozdelenie), veľkosť stránok verzus počet požiadaviek na stránku (Paretovo rozdelenie). Najčastejšími prenášanými súbormi na webe boli malé obrázky. Takisto sa potvrdilo, že dokumenty na webe sa veľmi často menia.

Pri skúmaní správania sa používateľov zostrojili základný model surfovania po webe, ktorý využíva čas prehliadania stránky, popularitu stránky a charakteristiky webovej stránky. Pomocou tohto modelu objavili, že počet klikov používateľov na webovej stránke sa riadi inverzným Gaussovým rozdelením (pozri obrázok 13).

Väčšinu činnosti používateľov pri prehliadaní webu tvorila navigácia medzi stránkami, t. j. používatelia buď nasledovali odkazy na ďalšie stránky alebo sa vracali späť a iba malé percento stránok navštívili napísaním adresy do prehliadača. Zistenia z výskumu boli užitočné pri vylepšovaní algoritmov na načítavanie stránok a správu obľúbených stránok a používateľských záložiek.



Obrázok 13. Počet klikov na používateľa na stránkach Xeroxu v máji 1998 [6].

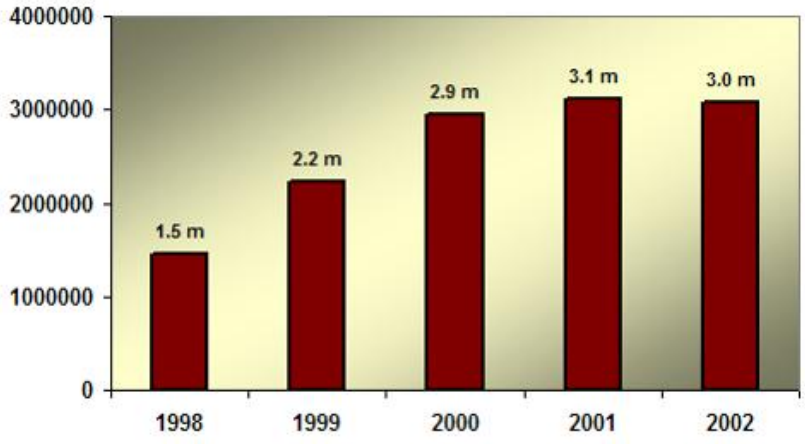
4.2 Výskum OCLC zameraný na charakterizovanie webu

Druhý výskum zameraný na charakterizovanie webu – *Výskum OCLC zameraný na charakterizovanie webu* (angl. *Web Characterization Research*) [7] pochádza z inštitúcie OCLC (Online Computer Library Center). Prebiehal v rokoch 1998 až 2002 a bol zameraný na vývoj verejne prístupného webu. Ročne analyzoval vzorky webu získané pripájaním sa na port 80 náhodne generovaných IP adries. Dnes by táto metóda nezachytila obrovské množstvo webových sídiel, ktoré používajú virtuálny webhosting (viacero domén na tej istej IP adrese). Do vzorky zahrnuli len verejné sídla alebo sídla, ktorých väčšina obsahu bola verejne prístupná. Výskum sledoval rôzne trendy vo vývoji webu ako:

- veľkosť a rast,
- internacionalizácia,
- používanie metaúdajov,
- popularita stránok.

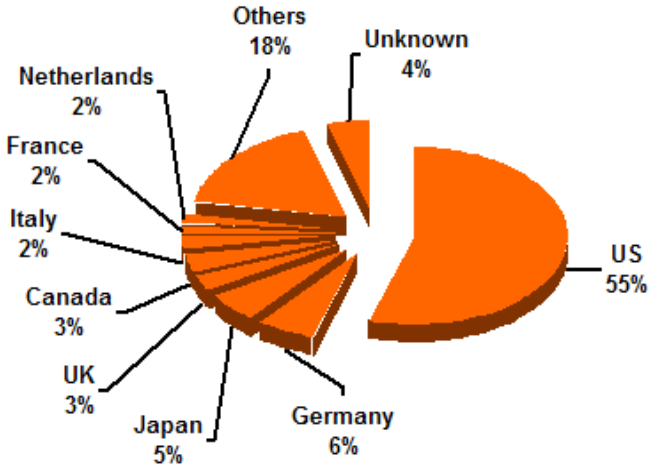
Podľa výsledkov výskumu obsahoval web v roku 2002 cez 3 milióny verejných webových sídiel. Podiel verejných lokalít bol vyhodnotený ako 35 % z celého webu. Jedno sídlo sa skladalo priemerne zo 441 webových stránok. Počas piatich rokov výskumu sa veľkosť verejného webu zdvojnásobila (pozri obrázok 14), jeho rast sa však postupne zastavoval. Autori uvádzajú, že pokles verejných sídiel bol spôsobený úbytkom ľudí a organizácií, ktorí si vytvárali a udržovali webové sídlo. Hoci počet verejných sídiel klesal, veľkosť webu stále rástla a zaznamenali nárast v priemernom počte stránok na jedno webové sídlo počas piatich rokov z 413 na 441.

V posledných rokoch web zaznamenáva obrovský rast a predpokladá sa, že v roku 2014 dosiahne počet webových sídiel miliardu [8].



Obrázok 14. Veľkosť verejného webu počas piatich rokov OCLC výskumu [7].

Z hľadiska internacionalizácie sa výskum zameriaval na zastúpenie krajín na webe. Zachytené webové sídla priradili k ich vlastníkom a ku krajinám, z ktorých pochádzali. Vzorky webu z roku 1999 ukázali, že polovica verejného webu pochádzala z USA a ostatné krajiny mali nanajvýš 5% zastúpenie. Na konci výskumu v roku 2002 podiel webu z USA stúpol na 55 %, zatiaľ čo zastúpenie ostatných krajín zostalo približne rovnaké (pozri obrázok 15). Celkovo bolo identifikovaných 76 krajín. Z týchto zistení autori usúdili, že najpoužívanejším jazykom na webe je anglický jazyk.



Obrázok 15. Rozdelenie vzorky verejného webu z roku 2002 podľa krajiny pôvodu [7].

Po preskúmaní jazykového zastúpenia vo vzorkách webu sa potvrdilo, že skutočne skoro tri štvrtiny webu boli v anglickom jazyku. Iba 7 % webových sídiel obsahovalo viacjazyčné verzie svojich stránok. V rámci výskumu porovnali jazykové zastúpenie vo vzorke verejného webu s jazykovým zastúpením bibliografických zdrojov z katalógu *WorldCat* (the OCLC Online Union Catalog), ktorý obsahuje okolo 45 miliónov zdrojov. Zistilo sa, že jazykové zastúpenie bolo približne rovnaké s väčšinovým zastúpením anglického jazyka.

V súčasnosti vedie v krajinnom zastúpení stále USA, no pribudlo viac obsahu z Európy a iných kontinentov [9]. Anglický jazyk sa používa v cca 56 % webu [10].

Počas piatich rokov výskumu sledovali trend v používaní metaúdajov na webe. Metaúdaje slúžia na opis zdroja informácií – webovej stránky. Na ich zápis vytvorili v HTML jazyku špeciálnu značku *meta*, ktorá obsahuje dva atribúty – názov opisovanej vlastnosti a jej hodnotu, napr. kľúčové slovo, autor a pod. Vo vzorkách webu skúmali výskyt práve tejto značky. Autori zistili, že metaúdaje sa čím ďalej, tým viac používajú. Nárast pripisujú aj automatickému vkladaniu tejto značky v HTML editoroch. Značky sa však nevyužívali na podrobný opis webu. Jedna stránka obsahovala priemerne len dve až tri značky. Ukázalo sa taktiež, že autori stránok nie sú príliš ochotní si osvojiť formálne schémy metaúdajov (napr. Dublin Core), ktorými by mali opísať svoju stránku a len necelé percento využilo formálnu schému.

Ročne sa vyhodnocovali aj prepojenia medzi stránkami a vytvárali štatistiky najodkazovanejších verejných stránok. V roku 2002 bola najodkazovanejšou stránkou stránka *www.adobe.com*. Bolo to pravdepodobne kvôli odkazom na softvér *Adobe Flash Player* a *Adobe Reader*, ktoré sú dnes už súčasťou webových prehliadačov. Ostatné stránky boli väčšinou spravodajské weby alebo mailové klienty.

V súčasnosti sú najdokazovanejšie stránky služby sociálnych sietí, blogy a vyhľadávače (pozri tabuľku 1). Rôzne analytické spoločnosti dnes udržiavajú štatistiky o popularite stránok založené buď na počte odkazujúcich stránok alebo návštevnosti stránok (napr. Moz [12], Alexa [13]). Polovica najodkazovanejších stránok sú aj najnavštevovanejšie stránky [13]. Práve tieto stránky sa nachádzajú v strede motýlikovej štruktúry webu opísanej v predchádzajúcej kapitole.

Tabuľka 1. Najodkazovanejšie stránky z roku 2002 [11]. z roku 2013 [12] a z roku 2014[3].

	2002	2013	2014
1.	adobe.com	facebook.com	google.com
2.	microsoft.com	twitter.com	facebook.com
3.	geocities.com	google.com	yotube.com
4.	netscape.com	youtube.com	yahoo.com
5.	members.aol.com	wordpress.com	baidu.com
6.	yahoo.com	adobe.com	wikipedia.org
7.	amazon.com	blogspot.com	qq.com
8.	google.com	wikipedia.com	twiter.com
9.	macromedia.com	wordpress.com	tabao.com
10.	cnn.com	linkedin.com	ymazon.com

[3] http://www.alexa.com/topsites

4.3 Ako dynamický je web?

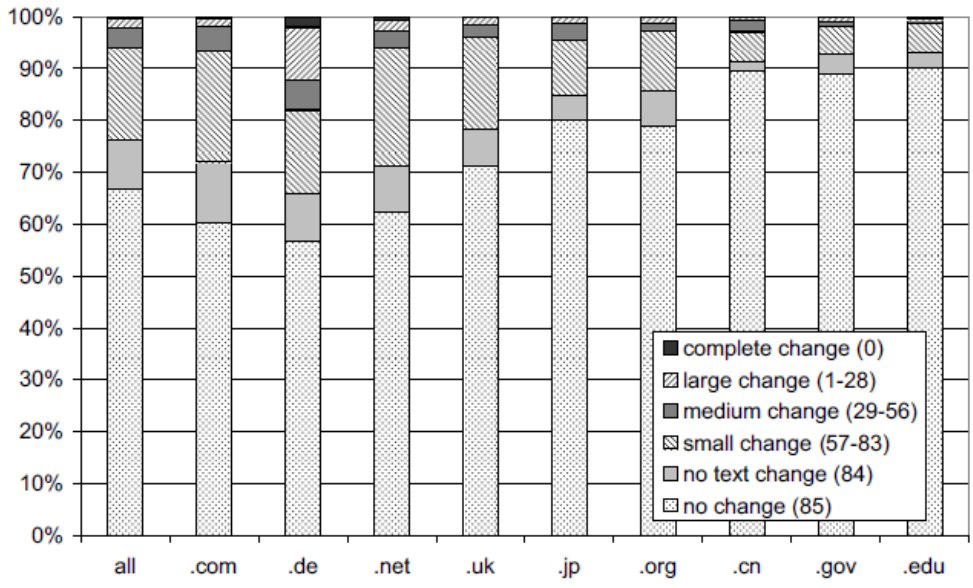
V roku 2004 vznikli dve štúdie, ktoré sa zaoberali dynamickosťou webu, t. j. ako sa menil web v čase. Obe štúdie analyzovali týždenne súbor stránok a vyhodnocovali zmeny oproti minulým obdobiam. Zameriavali sa na tieto aspekty:

- zmeny na stránkach,
- vytváranie, zanikanie stránok a evolúcia štruktúry prepojení stránok.

4.3.1 Zmeny na webových stránkach

Fetterly [14] vo svojej štúdii počas 11 týždňov analyzoval 150 miliónov webových stránok a vytváral týždenné prehľady, ktoré potom porovnával. Zameriaval sa na sledovanie zmien na stránkach – to, ako rýchlo sa menia, aké sú najčastejšie zmeny a či zmeny súvisia s nejakými inými vlastnosťami webovej stránky.

Výsledky štúdie uvádzajú, že stránky zvyčajne menili len HMTL zápis alebo vykonávali malé zmeny v obsahu. Štúdia sledovala súvislosť medzi doménou najvyššej úrovne, na ktorej sídlila stránka a frekvenciou zmien stránky (pozri obrázok 16). Zistenia ukázali, že existuje silný vzťah medzi doménou najvyššej úrovne a frekvenciou zmien, ale slabý vzťah medzi doménou najvyššej úrovne a rozsahom zmien.



Obrázok 16. Podiely zmien na webových stránkach podľa domény najvyššej úrovne [14].

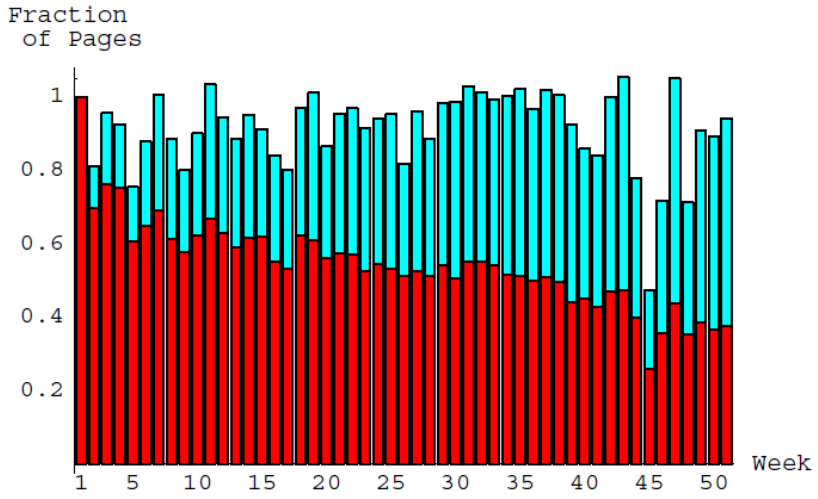
Ďalším sledovaným faktorom bola veľkosť stránky. Veľkosť väčšiny stránok (63 %) sa pohybovala v rozmedzí 4-32KB (dnes je priemerná veľkosť HTML zápisu stránky 55KB a celej stránky až 1576KB [15]). Prekvapilo, že veľké stránky sa menili oveľa viac a oveľa častejšie ako malé stránky. Najmenej meniace sa stránky boli vládne (.gov) a z domény vzdelávania (.edu) oproti komerčným doménam (.com, .net).

Významným prínosom štúdie bolo zistenie, že zo zmien na stránkach v minulosti sa dajú predpovedať budúce zmeny. Tento fakt sa dá prakticky využiť vo webových preliezačoch.

4.3.2 Evolúcia stránok a prepojení medzi nimi

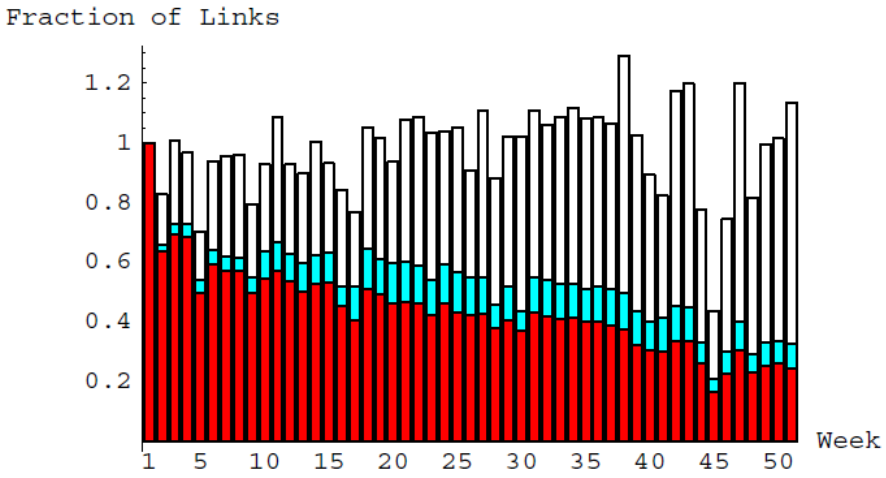
Ntoulas [16] vypracoval štúdiu, v ktorej počas jedného roka sledoval, ako sa týždenne mení obsah a štruktúra prepojení stránok 150 webových sídiel. Cieľom tejto štúdie bolo získať prehľad o tom, ako by sa mali webové vyhľadávače vyrovnať s neustálym vývojom webu. Sledoval najmä počty vytvorených a zaniknutých stránok a prepojení.

Vo výsledkoch sa uvádza, že staré stránky sa rýchlo nahradzovali novými, ale ich obsah sa veľmi podobal už existujúcim stránkam. Týždenne zaznamenali prírastok nových stránok o 8%. Stránky, ktoré sa zachovali dlhší čas, prechádzali len malými zmenami. Autori predpovedali, že ak bude trend vývoja webu pokračovať, iba 20% stránok, ktoré existujú na začiatku roka, bude dostupných aj o rok (pozri obrázok 17). Podobne ako v predchádzajúcej štúdii, aj tu autori prišli k záveru, že zmeny na stránkach, ktoré sa často menia, možno predpovedať na základe ich zmien v minulosti.



Obrázok 17. Podiel stránok z prvého týždňa, ktoré stále existovali po *n* týždňoch (červené, resp. tmavé časti) a nových stránok (modré, resp. svetlé časti) [16].

Pri analyzovaní prepojení medzi stránkami autori zistili, že prepojenia sa menia ešte rýchlejšie ako stránky a väčšina z nich pretrvá menej ako pol roka. Každý týždeň vzniklo 25 % nových prepojení. O rok bolo 80 % prepojení, ktoré existovali na začiatku, nahradených novými (pozri obrázok 18).



Obrázok 18. Podiel prepojení medzi stránkami z prvého týždňa, ktoré stále existovali po *n* týždňoch (spodné červené, resp. tmavé časti), nových prepojení z pôvodných stránok (stredné časti) a nových prepojení z nových stránok (horné časti) [16].

4.3.3 Mŕtve linky: problém 404

Problém 404 veľmi súvisí s evolúciou webu. V dôsledku neustáleho zanikania a nahradzovania stránok sa mnohé prepojenia stávajú neplatnými, tzv. mŕtvymi, pretože odkazujú na už neexistujúci obsah. Za posledných takmer dvadsať rokov výskumu v oblasti charakterizovania webu sa tomuto problému venovalo viacero autorov, ktorí skúmali perzistenciu buď webových stránok alebo prepojení medzi nimi.

Priemerný čas existencie stránky bol v roku 1997 podľa Kahleho 44 dní [17]. V roku 2001 to bolo podľa [18] 75 dní. V roku 2003 bol podľa Kahleho priemerný čas už 100 dní [19]. Z uvedených výskumov vyplýva, že priemerný čas existencie stránky sa predlžuje. Nenašli sme však žiadne štúdie, ktoré by určili priemerný čas existencie stránky dnes.

Keďže analyzovať celý web je veľmi zložité kvôli jeho rozľahlosti, viaceré výskumy sa vykonali na nejakej podmnožine webu, napríklad na knižničných zdrojoch. Už v roku 2002 konštatovali v OCLC, že knižničné zdroje zodpovedajú v niektorých ohľadoch webu (napr. jazykové zastúpenie zdrojov) [7]. Autori [18] prišli s výsledkom, že po piatich rokoch 23-53 % prepojení z CiteSeer článkov už neplatilo. V [21] sa uvádza, že 27 % prepojení z CACM/Computer článkov bolo po piatich rokoch neplatných. Výsledky [22] udávajú polčas premeny prepojení v D-Lib Magazine článkoch 10 rokov. Údaje z týchto troch nezávislých výskumov na troch odlišných knižničných zdrojoch sú približne rovnaké (cca 25% znehodnotenie prepojení za 5 rokov). Ale z porovnania so štúdiou [20], ktorá vyhodnocovala rovnaký parameter na údajoch zo všeobecného webu s výsledkom znehodnotenia až 67% prepojení za štyri roky, usudzujeme, že knižničné zdroje a prepojenia v nich vykazujú väčšiu perzistenciu ako zdroje na všeobecnom webe. Najaktuálnejší výskum [23] sa zaoberal perzistenciou prepojení v príspevkoch na službe sociálnych sietí Twitter s výsledkom 11%

znehodnotenia prepojení po jednom roku, čo je v porovnaní s knižničnými zdrojmi oveľa väčšia miera znehodnotenia.

4.4 Blogosféra

Od roku 2003 sledujeme rozmach blogov na webe. Pojem blog vznikol v roku 1997 zo slova *weblog*. Blog slúži na vyjadrenie myšlienok a názorov jedinca na webe v tvare spravidla pravidelných (denných, týždenných) záznamov, ktoré sa zvyčajne zoraďujú v opačnom chronologickom poradí. Blog sa vo veľkom využíva aj na propagáciu výrobkov a reklamu. Blogosféra je časť webu tvorená webovými doménami, ktoré obsahujú stránky, na ktorých sa blogy zverejňujú. Podľa [24] sa veľkosť blogosféry od roku 2003 zdvojnásobí každých 6 mesiacov.

V súčasnosti existuje cez 170 miliónov blogov. Množstvo ľudí bloguje prostredníctvom služieb sociálnych sietí – ide najmä o tzv. mikroblogy (napr. Twitter, Facebook). Motiváciou písať blog môže byť zábava, ale aj peniaze, ktoré dokáže blogger zarobiť, ak zverejní na svojom navštevovanom blogu reklamu alebo propaguje v blogu nejaký výrobok. Najviac zarábajúce blogy sú The Huffington Post a Mashable. Okolo 14% bloggerov sa živí len blogovaním a priemerne ročne zarobia okolo 24 tisíc dolárov. Viac ako polovica bloggerov, ktorých platia za každý príspevok, ale nezarobí ani tisíc dolárov za rok [25].

4.5 Zhrnutie

Najvýznamnejšie štúdie o charakterizovaní webu boli výskum komunity W3C *Web Characterization Activity* a OCLC *Web Characterization Research*. Hoci majú už cez desať rokov, priniesli výsledky, ktoré platia doteraz. Najväčším prínosom bolo zavedenie jednoznačnej terminológie a definície pojmov webu komunitou W3C. Bol to základ pre rozvoj ďalšieho výskumu v oblasti webu. V súčasnosti nie sú známe také rozsiahle štúdie zrejme kvôli veľkosti webu a náročnosti až nemožnosti jeho celého spracovania a analyzovania. Okrem toho nie všetok obsah na webe tvorí text. Veľkú časť webu tvorí audiovizuálny obsah. S týmito problémami sa potýkali už prvé štúdie. Dnes je web mnohonásobne väčší. Jeho celé spracovanie by neumožnila ani jeho dynamická povaha. Pokiaľ by sme analyzovali web ako celok, jeho tvar by bol už celkom iný.

Medzi najzaujímavejšie trendy dnes patrí nárast blogov a mikroblogov a masívne používanie služieb sociálnych sietí, kde sa sústredí veľká časť činnosti používateľov na webe. Takéto stránky patria medzi najnavštevovanejšie a najviac na ne odkazujú. Tvoria jadro motýlikovej štruktúry webu.

Literatúra

[6] Pitkow, J. E.: Summary of WWW Characterizations. In *World Wide Web*, vol. 2, no. 1-2, (1999), pp. 3-13.

[7] O'Neil E. T., Lavoie, B. F., Bennet, R.: Trends in Evolution of the Public Web: 1998-2002. In *D-Lib Magazine*, vol. 9, no. 4, (2003).

[8] November 2013 Web Server Survey, (2013). Dostupné na: http://news.netcraft.com/archives/2013/11/01/november-2013-web-server-survey.html

[9] The US hosts 43% of the world's top 1 million websites. (2012), Dostupné na: http://royal.pingdom.com/2012/07/02/united-states-hosts-43-percent-worlds-top-1-million-websites/

[10] Usage of content languages for websites. (2013). Dostupné na: http://w3techs.com/technologies/overview/content_language/all

[11] Web Characterization Stats – Linkage patterns. (2003). Dostupné na: http://www.oclc.org/research/activities/wcp/stats/linkage.html

[12] The Moz Top 500. (2013). Dostupné na: http://moz.com/top500

[13] The top 500 sites on the web. (2013). Dostupné na: http://www.alexa.com/topsites

[14] Fetterly, D., Manasse, M., Najork, M. et al.: A Large-scale Study of the Evolution of Web Pages. In *Proceedings of the 12th International Conference on World Wide Web*, (2003), pp. 669-678.

[15] HTTP Archive: Intersting Stats. (2013). Dostupné na: http://httparchive.org/interesting.php?a=All&l=Nov%201%202013

[16] Ntoulas, A., Cho, J., Olston, Ch.: What's New on the Web?: The Evolution of the Web from a Search Engine Perspective. In *Proceedings of the 13th International Conference on World Wide Web*, ACM Press, (2004), pp. 1-12.

[17] Kahle, B.: Preserving the Internet. In *Scientific American*, vol. 276, no. 3, (1997), pp. 82-83.

[18] Lawrence, S., Pennock, D. M., Flake, G. W. et al.: Persistence of Web References in Scientific Research. In *Computer*, vol. 34, no. 2, (2001), pp. 26-31.

[19] Weiss, R.: On the Web, Research Work Proves Ephemeral. In *Washington Post*, November 24, (2003), pp. A08.

[20] Koehler, W.: A longitudinal study of Web pages continued: a consideration of document persistence. In *Information Research*, vol. 9, no. 2, (2004).

[21] Spinellis, D.: The Decay and Failures of Web References. In: *Communications of the ACM*, vol. 46, no. 1, (2003), pp. 71-77.

[22] McCown, F., Chan, S., Nelson, M. L. et al.: The Availability and Persistence of Web References in D-Lib Magazine. In: *Proc. of the 5th Int. Web Archiving Workshop and Digital Preservation (IWAW'05)*, (2005).

[23] SalahEldeen, H., Nelson, M., L.: Losing My Revolution: How Many Resources Shared on Social Media Have Been Lost? In: *Proc. of the Second International Conference on Theory and Practice of Digital Libraries*, Springer Verlag, (2012), pp. 125-137.

[24] Sifry, D.: State of the Blogosphere. (2006). Dostupné na: http://www.sifry.com/alerts/archives/000432.html

[25] McGrail, M.: The Blogconomy: Blogging Stats [INFOGRAPHIC]. (2013). Dostupné na: http://socialmediatoday.com/mikevelocity/1698201/blogging-stats-2013-infographic

5 Archivovanie webu

Webové stránky majú zrýchlený životný cyklus. Mnoho z nich nenávratne zaniká už po veľmi krátkej dobe. Pritom na webe sú informácie, ktoré tvoria digitálne dedičstvo spoločnosti. O dlhodobé uchovanie týchto údajov sa starajú iniciatívy archivovania webu, ktorých potreba v posledných rokoch začína rásť. Tieto iniciatívy vytvárajú verejne dostupný archív stránok pre lokálne oblasti, ale taktiež sú tu organizácie archivujúce celý web. Za týmto účelom navrhli niekoľko spôsobov zhromažďovania informácií z webu na jedno miesto a implementovali mnoho nástrojov umožňujúcich archivovanie.

Archivovanie webu je proces zhromažďovania webových stránok a informácií, ktoré obsahujú, a ich trvalého uchovávania v archíve. Ide o proces podobný tradičnému archivovaniu papierových dokumentov. Informácie sa vyberú a uložia tak, aby boli dostupné verejnosti. O webové archívy sa starajú weboví archivári. Keďže ide o ukladanie masívneho množstva informácií, archivári využívajú automatizované procesy. Softvér, ktorý umožňuje zber stránok, je známy ako *preliezač*. Preliezače cestujú po webe a zároveň kopírujú a ukladajú informácie. Archivované stránky sú potom dostupné online a môžu sa prezerať a čítať. Ide však len o snímku stránky v určitom časovom bode a nie je možné na nej vytvárať dopyty [8].

Existujú dva pohľady na archivovanie webu. Mikro-archivovanie je uchovanie jednej stránky individuálnym subjektom, zatiaľ čo makro-archivovanie sa deje vo veľkom meradle nejakou archívnickou iniciatívou [6]. Tu opíšeme archivovanie predovšetkým z druhého spomenutého pohľadu.

5.1 Dôvody archivovania

Web je extrémne prchavý. Väčšina informácií sa nenávratne stratí po krátkej dobe. Až 80% stránok sa po roku zmení alebo zmizne. Spoločnosť tak prichádza o digitálne dedičstvo, ktoré pozostáva napríklad z historických a vedeckých informácií. Tlačené publikácie sa degradujú, ak citujú

zdroje dostupné online. Ľudia prichádzajú o svoje spomienky pri strate fotografií atď. Nedostupné linky majú dopad na rôzne aplikácie, ktoré ponúkajú vyhľadávanie či zdieľanie záložiek. Web tak potrebuje iniciatívy, ktoré bojujú s jeho prchavosťou. Malo by sa zabezpečiť, aby informácie zostali natrvalo uložené, aby mohli poskytnúť vedomosti budúcim generáciam [3].

Z pohľadu vlastníka stránky je archivovanie dôležité napríklad z právnych dôvodov. Informácia na stránke má rovnaké právne postavenie ako písaný ekvivalent. Organizácie tak musia byť pripravené na argumenty, čo bolo a čo nie uverejnené na stránke a vytvárať tak vlastné archívy [5].

5.2 Problémy pri archivovaní

Archivovanie webu nie je jednoduchou záležitosťou. Narážame pri tom na zložité technické ako aj právne a organizačné problémy.

Z technického hľadiska je nemožné, aby jedna organizácia bola schopná archivovať celý web, ktorý je obrovský a rýchlo narastá. Je preto potrebné, aby z dlhodobého hľadiska išlo o činnosť založenú na spolupráci. Okrem viditeľných statických stránok je tu 400 až 500 krát väčší obsah skrytého alebo hlbokého webu, ktorý pozostáva zo stránok vytváraných dynamicky na základe databázy, alebo zo stránok, ktoré sa chránia prihlasovacími údajmi. Ďalším technickým problémom je rýchlosť, s akou sa stránky strácajú. Archivovaním sa nemusia stránky s krátkym životným cyklom vôbec stihnúť uložiť.

Ďalší typ problémov je spätý s autorským právom či zodpovednosťou za obsah. V mnohých krajinách je právne prostredie nevďačné alebo dokonca nehostinné k webovým archívom. Najbezpečnejšou cestou ako prekonať tieto problémy, je opatrne vyberať zdroje, ktoré sa budú archivovať a vytvoriť efektívnu politiku manažovania práv a efektívnych procesov pre mazanie určitého obsahu.

Z organizačného hľadiska neexistuje jedna organizácia, ktorá zodpovedá za web. Ten sa vyvíja decentralizovaným spôsobom. Nie je tu teda správny orgán, ktorý prikáže štandardy pre uchovávanie webu. Za obsah stránky zodpovedá každý vlastník sám. Preto existujú viaceré organizácie, ktoré sa podieľajú na archivovaní webu. Väčšinou sa zameriavajú len na nejakú podmnožinu webu svojho záujmu, ako napríklad národný archív krajiny, ktorý sa zaujíma o archivovanie stránok spadajúcich do jej domény. Tento prístup môže byť užitočný, ale bola by potrebná koordinácia, aby nedochádzalo k duplicite a aby z archívov pre používateľa nevznikol iba neusporiadaný zhluk repozitárov [2, 7].

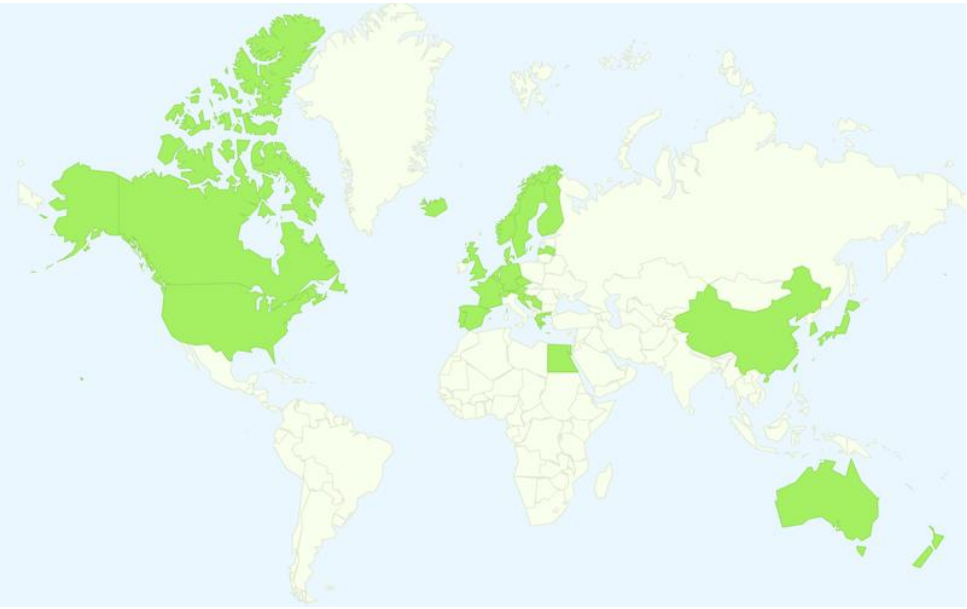
5.3 Iniciatívy archivovania

V súčasnosti vraj existuje 67 organizácií, ktoré sa venujú archivovaniu webu. Iniciatívy, ktoré sa venujú archivovaniu na celosvetovej úrovni sú *Internet Archive*, *Internet Memory* a *California Digital Library*. Najstaršou iniciatívou, ktorá vznikla ešte v roku 1996, je nezisková organizácia Internet Archive, ktorá sídli v San Franciscu (USA). Služba, ktorú poskytuje, sa nazýva Archive-it. Organizácia Internet Memory vznikla v roku 2004, sídli vo Francúzsku a Holandsku

a poskytuje službu ArchiveTheNet. California Digital Library vznikla o rok neskôr a jej služba sa volá Web Archiving Service. Služby ako tieto tiež využívajú iné organizácie, ktoré nemôžu manažovať svoje vlastné archívy. Tieto tri archívy spolu zamestnávajú 35 zamestnancov na plný úväzok (pre jednotlivé organizácie v uvedenom poradí je to 12, 21 a 4).

Drvivá väčšina iniciatív (až 80%) výlučne archivuje obsah príbuzný krajine, regiónu alebo inštitúcií, v ktorej sa nachádza. Okrem toho sú však aj také, ktoré uchovávajú obsah stránok pre vybrané zahraničné krajiny, ako napríklad pre juhoamerické krajiny (Latin American WA) alebo krajiny v Pacifiku (WA Pacific Islands). Na obrázku 19 sú zvýraznené krajiny, v ktorých sa nachádzali v roku 2012 iniciatívy archivovania. Spolu je týchto krajín 23.

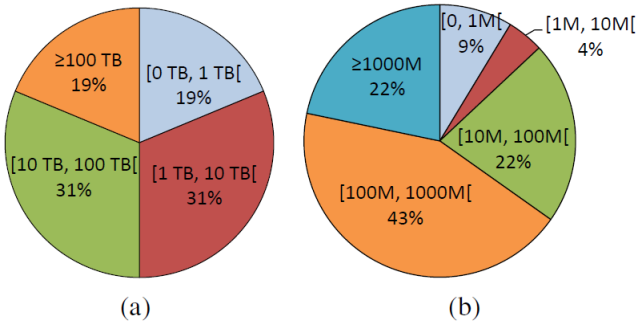
Všetky organizácie spolu v roku 2011 zamestnávali 112 zamestnancov na plný úväzok a 166 na čiastočný. Pre porovnanie, firma Google zamestnávala v tom čase 24 400 zamestnancov [3].



Obrázok 19. Krajiny, v ktorých sa nachádzajú archivujúce iniciatívy (2012) [3].

5.4 Miera doposiaľ archivovaného webu

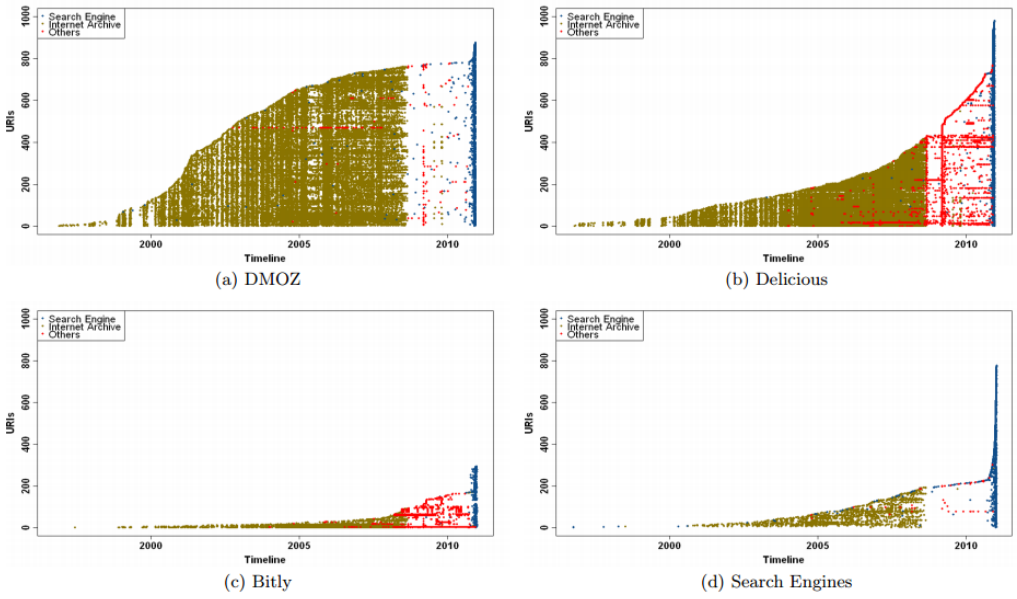
Gomes vo svojej práci z roku 2011 [3] preskúmal veľkosť kolekcií jednotlivých archívnickych iniciatív. Veľkosti jednotlivých kolekcií sú znázornené v grafe na obrázku 20. Výsledky ukázali, že 50% archívov je menších ako 10TB a že tri štvrtiny kolekcií pozostávajú z menej ako miliardy objektov (objekt zodpovedá HTML stránke, obrázku na stránke atď.). Internet Archive zozbieral 150 000 miliónov objektov od roku 1996 z celkového množstva 182 000 miliónov (1.8×10^{11}) objektov. Pre porovnanie Google v roku 2008 oznámil, že jediný snímok webu pozostáva z trilióna (10^{12}) URL adries.



Obrázok 20. Veľkosti archivovaných kolekcií: objem údajov v TB (a), množstvo objektov (b) [1].

V podobnej práci z roku 2011 sa pokúsil Ainsworth [1] odhadnúť percentuálne pokrytie archivovaného webu. Experiment bol nastavený tak, že vybrali vzorku 1000 webových adries z rôznych služieb (DMOZ, Delicious, Bitly a najväčšie vyhľadávače). Potom nechali tieto adresy vyhľadať pomocou služby Memento, ktorá agreguje webové archívy.

Obrázok 21 znázorňuje rozloženie archivovaných verzií stránok v čase. Farba bodiek určuje archív, ktorý danú stránku obsahuje. Najväčšia iniciatíva Internet Archive poskytuje veľké množstvo verzií aj najväčšiu históriu. Medzi tým, ako stránku uložili a poskytli archívom verejnosti, je istý časový rozdiel (6-24 mesiacov). Naopak vyhľadávače ako Google, Bing a Yahoo ukladajú najaktuálnejšie verzie stránok, avšak iba jednu pre jednu stránku, a po mesiaci ich väčšinou zmažú. Výsledky ukázali, že zatiaľ čo na vzorke získanej z DMOZ a Delicious bola vytvorená aspoň jedna záloha u 90% adries, na vzorke z Bitly to bolo len 30%. Nie je teda ľahké odhadnúť mieru archivovania webu, ale odhaduje sa, že sa pohybuje medzi 30% až 90%.



Obrázok 21. Graf distribúcie archivovaných verzií stránok na štyroch vzorkách (DMOZ, Delicious, Bitly, vyhľadávače). Hnedé bodky reprezentujú záznamy v Internet Archive, modré vyhľadávače a červené iné archivujúce iniciatívy a služby [1].

5.5 Prístupy k archivovaniu využívané v praxi

Najpopulárnejším prístupom k archivovaniu je *archivovanie na strane klienta* pre jeho jednoduchosť a škálovateľnosť. Táto metóda umožňuje archivovanie akejkoľvek stránky, ktorá je voľne dostupná na webe. Programy, tzv. preliezače, imitujú interakciu používateľa s webovou stránkou, začínajúc na nejakej adrese nasledujú linky a ukladajú otvorené stránky až kým nedosiahnu nejakú hranicu.

Archivovanie na strane servera zahŕňa priame kopírovanie súborov zo servera. Tento prístup sa môže použiť iba v spolupráci s vlastníkom stránky. Problémom je umožnenie prehliadania takto uloženého obsahu, ak sa stránky generujú dynamicky. Je potrebné prostredie s takými istými parametrami ako na serveri, aby fungoval prístup k databáze, skripty, šablóny. Výhodou je možnosť archivovať časti webu skryté pred preliezačmi.

Archivovanie založené na transakciách funguje taktiež na serveri a vyžaduje k nemu prístup, takže je potrebná spolupráca s vlastníkom stránok. Zaznamenávajú sa transakcie medzi používateľom a serverom. Tento prístup je zaujímavý tým, že obsah, ktorý nikdy nezobrazili, nearchivujú. Výhodou je teda, že zaznamená sa len to, čo niekto videl [7].

5.6 Technológie, nástroje a služby pre archivovanie

Archivovanie webu podporujú viaceré voľne dostupné nástroje, ktoré umožňujú získanie obsahu, jeho ukladanie, navigovanie a vyhľadávanie vo vytvorenom archíve. Pre bežných používateľov sa zverejnili služby na prezeranie veľkých archívov vytváraných iniciatívami archivovania.

5.6.1 Získanie obsahu

Nástroje na získanie obsahu webu sú preliezače, ktoré majú špeciálne možnosti extrahovania polí alebo obsahu z webových stránok. Štyrmi najznámejšími nástrojmi sú Heritrix, HTTrack, Wget a DeepArc.

Heritrix[4] je open-source preliezač, ktorý vytvoril Internet Archive v roku 2003 a odvtedy ho udržuje a využíva na sťahovanie vo veľkom meradle. Stiahnuté zdroje sa ukladajú vo formáte Arc alebo štandardizovanom formáte WARC. Formát Arc ukladá viaceré zdroje v jedinom súbore, aby sa predišlo udržiavaniu veľkého množstva malých súborov. Ako rozhranie možno použiť Web Curator Tool, ktorý podporuje procesy ako oprávnenia, plánovanie úloh, hodnotenie kvality a zber opisných metaúdajov.

HTTrack[5] je voľne dostupnou utilitou, ktorá umožňuje stiahnutie celej stránky do lokálneho priečinka. Zachytí HTML kód a príslušné obrázky a iné súbory a následne rekurzívne buduje štruktúru priečinkov lokálne. Dokáže zaistiť správnosť štruktúry relatívnych odkazov, aby sa lokálna kópia dala prezerať ako online stránka. V prípade sieťového prerušenia automaticky zopakuje sťahovanie súborov. Dokáže aktualizovať stiahnuté stránky.

[4] https://webarchive.jira.com/wiki/display/Heritrix/Heritrix
[5] http://www.httrack.com/

GNU Wget[6] je program vytvorený v roku 1996, ktorý podporuje sťahovanie súborov cez HTTP, HTTPS a FTP protokoly. Má podobnú funkcionalitu ako HTTrack, nemá však grafické rozhranie [4].

5.6.2 Webové archívy

Jednoduchým typom archívu je *súborový systém*, kde sa odkazy na dokumenty konvertujú na relatívne. Je založený na archivovaní súborov. Hypertextová navigácia sa vykonáva v rámci lokálneho súborového systému. Využíva sa na archivovanie jednej stránky alebo len v malom meradle.

Pre archivovanie vo veľkom sa využívajú *archívy založené na odpovediach servera*. Odpovede zo servera sú uložené kontajnerových súboroch vo formáte *WARC*. Ide o štandardizovaný formát, ktorý agreguje viaceré zdroje do jedného súboru. Tento prístup vyžaduje webový server, ktorý vytiahne zo súborov obsah a zobrazí používateľovi. Schéma názvov (ako aj parametrov dynamických stránok) sa zachováva a umožňuje navigáciu po stránkach tak, ako ju preliezli [7].

5.6.3 Prehliadanie a vyhľadávanie vo webových archívoch

Na prehliadanie uloženého obsahu v podobe WARC súborov slúži nástroj *wayback*[7]. Ide o implementáciu aplikácie The Internet Archive Wayback Machine napísanú v jazyku java. Produkčnú verziu Wayback Machine implementovali v jazyku perl, pričom vznikla motivácia verejnej distribúcie aplikácie ako open-source a v roku 2005 vznikla prvá voľne dostupná verzia. Aplikácia dokáže pracovať vo viacerých módoch, ako samotná aplikácia na jednom uzle, tak aj ako distribuovaný systém s archívmi na viacerých uzloch. Zobrazenie, resp. vyhľadanie stránky sa uskutočňuje na základe URL adresy.

Textové vyhľadávanie vo webových archívoch umožňuje nástroj NutchWAX[8], ktorý beží na platforme Hadoop. Je potrebné vytvoriť index z WARC súborov, nad ktorým sa vykonávajú dopyty.

5.6.4 Archivovanie hlbokého webu

Nástroj DeepArc[9] vyvinula iniciatíva National Library of France na archivovanie databáz zo stránok, ktoré poskytujú prístup k digitálnym objektom (knihy, články, obrázky atď.) - tzv. dokumentárne brány. Opisy a identifikátory objektov sú uložené v relačnej databáze a samotné objekty sú uložené v súborovom systéme. DeepArc musí byť nainštalovaný na serveri vlastníka stránok, ktorý pomocou nástoja zmigruje štruktúru a obsah databázy na cieľový otvorený a štrukturovaný formát XML. Následne možno získať metaúdaje s príslušnými objektami zo stránky. Prístup k takejto XML databáze potom poskytuje nástroj Xing[10]. Vytvorili ho v rámci

[6] https://www.gnu.org/s/wget/
[7] http://archive-access.sourceforge.net/projects/wayback/
[8] http://archive-access.sourceforge.net/projects/nutch/index.html
[9] http://deeparc.sourceforge.net
[10] http://sourceforge.net/projects/xinq/

iniciatívy Austrálskej národnej knižnice. Umožňuje vytvorenie webovej aplikácie, pomocou ktorej je možné vyhľadávať a prehliadať databázu.

5.6.5 Archivačné služby

Existuje niekoľko služieb, ktoré poskytujú archivovanie stránok, aby používateľ nemusel naštudovať použitie vyššie spomenutých nástrojov. Archive.is[11] je neplatenou online službou pre archivovanie jednotlivých stránok aj všetkých stránok podľa zadaných kritérií. Služba WebCite[12] umožňuje výskumníkom, editorom žurnálov a vydavateľom natrvalo uložiť a získať odkazy na zdroje. Peeep.us[13] slúži na archivovanie aktuálne zobrazenej stránky. Výhodou tejto služby je, že používateľ si môže uložiť stránku chránenú prihlasovacími údajmi, nedá sa však automaticky uložiť viacero stránok.

5.7 Zhrnutie

Archivovanie webu sa pre spoločnosť stáva dôležitou úlohou, pretože na webe sa nachádza naše digitálne dedičstvo. Informácie, ktoré na webe pribúdajú, veľkou rýchlosťou aj zanikajú. O archivovanie webu na lokálnej, ale i globálnej úrovni sa starajú iniciatívy archivovania. Poskytujú služby pre verejnosť pre náhľad na webové stránky, ktoré už nie sú dostupné. Zdá sa, že zatiaľ sa archivuje len malá časť webu a pokiaľ sa nezvýši záujem o vykonávanie tejto úlohy, percentuálne bude rásť len pomaly. Úlohu archivovania podporujú viaceré voľne dostupné nástroje na získavanie obsahu z webu, vytváranie archívov a vyhľadávanie v nich.

Referencie

[1] Ainsworth, S. et al.: How Much of the Web Is Archived? In: *JCDL 2011*, (2011).

[2] Farrell S. et al.: *A Guide to Web Preservation*. UKOLN / ULCC, (2010).

[3] Gomes D., Miranda J., Costa M.: A survey on web archiving initiatives. In *Proceedings of the 15th International Conference on Theory and Practice of Digital Libraries: Research and Advanced Technology for Digital Libraries*, (2011), pp. 408-420.

[4] Guy, M.: Web Archiving: Tools for Capturing, (2010). Dostupné na: http://blogs.ukoln.ac.uk/jisc-bgdp/2010/07/28/web-archiving-tools-for-capturing/index.html

[5] Fitch K.: Web site archiving - an approach to recording every materially different response produced by a website. In *Ausweb 03*, (2003).

[6] Brügger, N.: *Archiving websites. General Considerations and Strategies*. The Centre for Internet Research, Aarhus, (2005). ISBN: 87-990507-0-6.

[7] Masanès, J: *Web Archiving*. Springer, (2006). ISBN-13: 978-3540233381.

[8] Web Archiving Guidance, (2003). Dostupné na: http://www.nationalarchives.gov.uk/documents/information-management/web-archiving-guidance.pdf

[11] http://archive.is
[12] http://www.webcitation.org
[13] http://www.peeep.us

6 Vyhľadávanie na webe

V čase informačného prebytku a zahltenia sa stalo vyhľadávanie podstatnou časťou nášho života. Cieľom tejto kapitoly je oboznámiť čitateľa so širším kontextom vyhľadávania informácií na webe ako samostatnej vednej disciplíny a predstaviť aktuálne smery výskumu, ako je personalizácia, prispôsobovanie sa kontextom používateľov či zohľadnenie sémantiky pri vyhľadávaní.

Za posledných dvadsať rokov sa web stal súčasťou našich každodenných životov a spolu s ním nevyhnutne aj vyhľadávanie na webe. V roku 2012 mala prístup na internet viac ako tretina svetovej populácie[14]. Dôležitosť vyhľadávania pritom ilustruje fakt, že z času, ktorý ľudia trávia pripojení na internet, venujú vyhľadávaniu 21%[15] (čo je o percento viac, ako čítaniu webového obsahu). Tento údaj možno vysvetliť zistením od Nielsena[16]: používatelia začínajú v 88% prípadov riešiť úlohy na webe práve pomocou webového vyhľadávača.

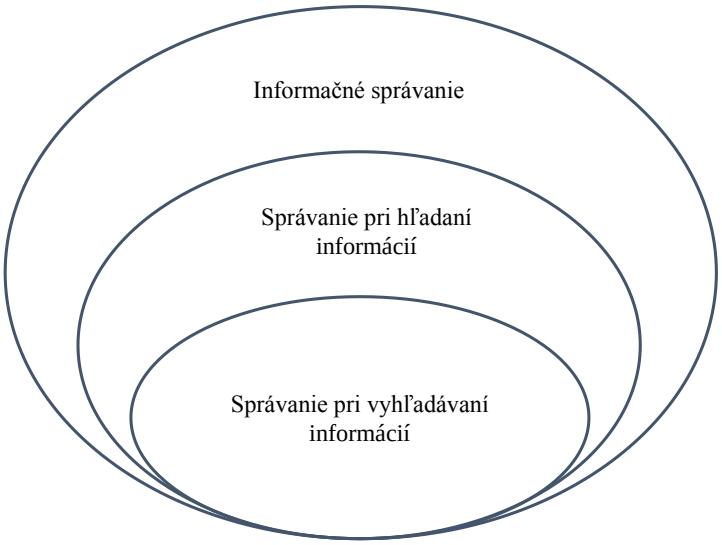
6.1 Terminológia a zasadenie do kontextu

V anglickej literatúre sa v súvislosti s vyhľadávaním môžeme stretnúť s viacerými, na prvý pohľad rovnoznačnými pojmami. *Vyhľadávanie na webe* (angl. *Web search*) možno chápať v užšom zmysle ako podmnožinu *vyhľadávania informácií* (angl. *information retrieval – IR*), ktorá súvisí s webovými vyhľadávačmi (angl. *search engine*). V širšom zmysle však možno vyhľadávanie chápať ako proces alebo ako správanie sa používateľov pri hľadaní.

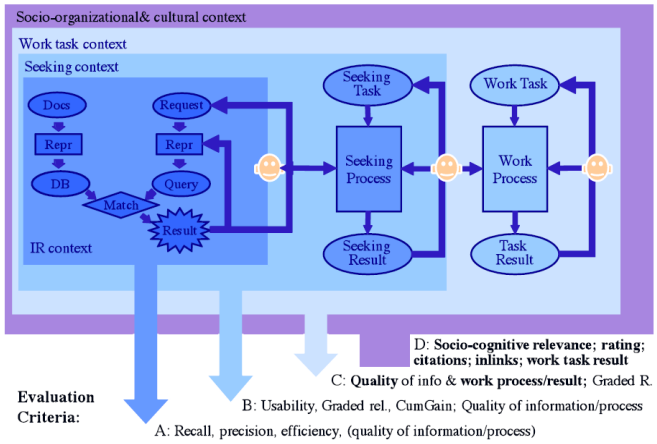
Ak kladieme dôraz na algoritmy, presnosť či úplnosť vyhľadávania, hovoríme väčšinou o vyhľadávaní informácií, ktoré sa zakladá na predpoklade, že pre zadaný dopyt používateľa existuje ideálna množina dokumentov – výsledkov, ktorej sa snaží čo najviac priblížiť. Naproti tomu, ak kladieme väčší dôraz na používateľa, jeho informačné potreby či širší kontext jeho vy-

[14] Internet World Stats, http://www.internetworldstats.com/stats.htm
[15] How People Spend Their Time Online: Infographic at Go-Gulf, http://www.go-gulf.com/blog/online-time/
[16] Nielsen, J.: When Search Engines Become Answer Engines, http://www.nngroup.com/articles/search-engines-become-answer-engines/

hľadávania (daný najčastejšie nejakou úlohou), hovoríme väčšinou o *hľadaní informácií* (angl. *information seeking – IS*). Hľadanie zahŕňa rôzne činnosti, ako učenie, porozumenie domény a pod., ktoré pri klasickom chápaní vyhľadávania informácií zanedbávame. Vyhľadávanie informácií tak často chápeme ako podmnožinu hľadania informácií [7, 24], ako môžeme vidieť na obrázkoch 22 a 23.



Obrázok 22. Vnorený model hľadania informácií [24].



Obrázok 23. Kontext vyhľadávania [7].

Navyše, na obrázku 23 vidíme, že aj hľadanie informácií môžeme zasadiť do širšieho kontextu pracovnej úlohy, kvôli ktorej sa hľadanie realizuje; tú zas môžeme chápať v spoločensko-organizačnom a kultúrnom kontexte daného používateľa.

6.2 História vyhľadávania na webe

História vyhľadávania na webe siaha do začiatku 90. rokov. Majitelia stránok zadávali svoje stránky do kategórií webových adresárov, t. j. do predpripravenej taxonómie. Príkladom webo-

vého adresára bol *Yahoo!* (*Yet Another Hierarchical Officious Oracle*)[17], ktorý vznikol roku 1994. Webové adresáre väčšinou zahŕňali len hlavnú stránku a nie jej podstránky. Keďže nedokázali udržať krok s exponenciálnym rastom obsahu na webe, v súčasnosti sa už masovo nepoužívajú, hoci podobné iniciatívy existujú dodnes, ako napr. *Open Directory Project*[18].

Webové adresáre postupom času nahradili vyhľadávačmi, ktoré sú založené na (invertovanom) indexe dokumentov, t. j. webových stránok. Stránky pritom nie je potrebné zadávať ručne, ale využívajú sa špecializované programy, tzv. *preliezače* (angl. *web crawler*), ktoré ich automaticky prechádzajú a sťahujú. Používatelia zadávajú dopyty pomocou postupnosti kľúčových slov, čo je v súčasnosti najrozšírenejšou paradigmou vyhľadávania. Jedným z prvých vyhľadávačov bol dnes už neexistujúci vyhľadávač *AltaVista* (vznikol v roku 1995). Najznámejším a najpoužívanejším vyhľadávačom súčasnosti je *Google*[19] (vznikol v roku 1998). Konkurovať sa mu snaží *Bing*[20] od Microsoftu (vznikol v roku 2009, predtým bol známy ako *Live Search*, resp. *MSN Search*).

6.3 Proces vyhľadávania

Vyhľadávanie možno opísať ako postupnosť týchto krokov [11]:

1. Používateľ zadá do vyhľadávača dopyt.
2. Vyhľadávač vráti zoznam výsledkov, z ktorých si používateľ niektorý vyberie.
3. Nasleduje fáza navigácie (surfovania), počas ktorej používatelia nasledujú odkazy na zvolenej stránke.
4. Ak medzi vrátenými výsledkami nie je žiadny, ktorý by vyhovoval požiadavkám používateľa, alebo ak sa zmenila jeho informačná potreba, používateľ upraví pôvodný dopyt a vracia sa na krok 1.

Každý zo spomínaných krokov pritom predstavuje samostatnú oblasť výskumu – existujú rôzne prístupy, ako podporiť zadávanie a tvorbu dopytu používateľmi (resp. ich úpravu, t. j. preformulovanie). Samostatnou oblasťou je, aké výsledky sa majú zobraziť – či zohľadniť len samotný dopyt, alebo aj ďalšie informácie o používateľovi a jeho záujmoch a upraviť (prispôsobiť, personalizovať) podľa nich zobrazené výsledky. Spôsob zobrazenia výsledkov môže byť tiež rôzny – od jednoduchého zoznamu odkazov s krátkymi súhrnmi, až po zložité 3D vizualizácie dokumentov a vzťahov medzi nimi.

6.3.1 Dopytovanie

Dopyt chápeme ako zhmotnenie (verbalizáciu) informačnej potreby používateľa. Ak chceme napr. zistiť, kto je prezidentom Spojených štátov amerických, je to naša informačná potreba, kto-

[17] V súčasnosti už pri vyhľadávaní tiež používa index dokumentov, ale jeho webový adresár možno nájsť na adrese: http://dir.yahoo.com/

[18] ODP: http://www.dmoz.org/

[19] https://www.google.com

[20] https://www.bing.com/

rú by sme mohli zhmotniť do dopytu „*usa prezident*“. Výsledkom takéhoto dopytu vo vyhľadávači Google by bol Barack Obama, ktorý je v čase písania tohto textu skutočne prezidentom USA, takže by sme boli v našom hľadaní úspešní. Pri podrobnejšom pohľade, pravda, to nie je až také jednoduché. Vyhľadávač totiž nevracia ako výsledok slovo, slová, tobôž nie vetu ani odpoveď, ale vracia zoznam dokumentov. Odpoveď na svoj dopyt si z toho, čo vráti vyhľadávač, musí zvedavec vytiahnuť sám. Na vyššie spomenutý dopyt, ktorý sme zadali 4. júla 2014, vrátil vyhľadávač okolo 438 miliónov „výsledkov“. Je však pravda, že už prvý z nich bol odkaz na webovú stránku, venovanú Barackovi Obamovi.

Broder vo svojej práci [3] rozdeľuje dopyty do troch skupín:

- *Navigačné* – ich cieľom je nájsť stránku na webe, o ktorej používateľ vie, že existuje.
- *Informačné* – cieľom je získať nejakú informáciu, o ktorej používateľ nevie, kde sa nachádza (a či vôbec), pričom môže byť roztrúsená aj na viacerých stránkach.
- *Transakčné* – cieľom používateľa je nájsť stránku, na ktorej bude prebiehať ďalšia interakcia (napr. nakupovanie na webe).

Analýzou dopytov pomocou dotazníkov a logov z vyhľadávača zistil, že navigačných dopytov je okolo 20-25%, informačných okolo 40-50% a transakčných okolo 30%. Tieto údaje majú význam najmä pri nejednoznačných dopytoch, keď potrebujeme rozpoznať, aká informačná potreba sa skrýva za daným dopytom.

Dopyty najčastejšie reprezentujeme a zadávame do vyhľadávačov slovne (preto hovoríme o verbalizácii informačnej potreby). Častým problémom je nejednoznačnosť dopytov (angl. *ambiguity*), ktorú spôsobuje viacvýznamovosť slov, resp. existencia slov, ktoré síce rovnako znejú, ale majú iný význam (*homonymá)*. Ešte vážnejším problémom je, že niekedy ani sám používateľ presne nevie, čo by chcel nájsť, alebo nedisponuje dostatočnou doménovou znalosťou, aby mohol sformulovať presný dopyt. Existujúce výskumy [5] ukazujú, že používatelia zadávajú najčastejšie krátke dopyty, ktoré málokedy prekročia dĺžku troch slov. Taktiež málokedy používajú pokročilé operátory. Naopak, ak spozorujeme, že ich používatelia začnú zadávať, resp. počet slov v ich dopytoch sa zvýši, indikuje to väčšinou, že majú problém danú úlohu vyriešiť [1]. Existujú tiež rozdiely medzi tým, aké dopyty zadávajú a ako sa správajú pokročilí používatelia (experti) a ako začiatočníci [20].

Kvôli spomínaným problémom pri tvorbe dopytov sa snažia tvorcovia vyhľadávačov používateľom pomôcť. Jedným z prístupov je automatické dopĺňanie dopytov na základe kontextu používateľa [2]. Iní sa snažia používateľov úplne odbremeniť od tvorby slovných dopytov, napr. pomocou fazetového vyhľadávania, vyhľadávania pomocou značiek alebo pomocou príkladov.

Fazetové vyhľadávanie [20] sa bežne používa v doménach, kde máme k dispozícii štruktúrované údaje (resp. metaúdaje, t.j. opisné údaje), napr. v online obchodoch, digitálnych knižniciach, a pod. Fazety predstavujú jednotlivé kategórie (napr. autor, rok vydania a pod. v prípade digitálnych knižníc), ktoré môžu nadobúdať rôzne hodnoty. Dopyt sa potom reprezentuje množinou zvolených hodnôt vybraných faziet.

Podobný prístup predstavuje *vyhľadávanie pomocou značiek (tagov)*. Na rozdiel od faziet, pri ktorých metaúdaje vznikajú automatizovane, resp. ich zadáva autor zdroja, tagy sú slovné značky, ktoré k zdrojom pridávajú samotní používatelia, aby sa k nim v budúcnosti vedeli ľahšie vrátiť. Často sa vizualizujú v podobe oblaku, kde sú jednotlivé značky odlíšené rôznou veľkosťou fontu alebo farbou podľa ich významnosti (relevancie) [12]. Dopyt sa potom reprezentuje postupnosťou zvolených značiek.

Dopytovanie pomocou príkladu (angl. *query by example*) si nachádza uplatnenie najmä v oblasti multimédií, ako sú napr. obrázky [17]. Dopytom je samotný dokument (napr. obrázok) alebo množina dokumentov (obrázkov) a výstupom je množina podobných dokumentov. Môžeme tiež uvažovať negatívne príklady; v takom prípade bude množina výsledkov čo najviac nepodobná zadaným dokumentom (dopytu).

Za zlatý grál pri tvorbe dopytu možno považovať otázky v prirodzenom jazyku – namiesto vymýšľania vhodného dopytu sa používatelia vyhľadávača spýtajú rovnako, ako by sa spýtali nejakého človeka. Ak uvažujeme príklad zo začiatku tejto podkapitoly, zadali by sme ako dopyt priamo otázku „*Ako sa volá súčasný prezident USA?*“. Aby vyhľadávač mohol na takto zadané otázky odpovedať, vyžaduje si to väčšinou využitie pokročilých metód spracovania prirodzeného jazyka a porozumenie zmyslu (sémantike), čomu sa čiastočne venujeme v časti 6.5.

6.3.2 Zobrazovanie výsledkov

Ďalšou veľkou oblasťou pri vyhľadávaní na webe je nepochybne zobrazovanie výsledkov. Najčastejšie sa môžeme stretnúť so zobrazením v tvare usporiadaného zoznamu, v ktorom sú jednotlivé webové stránky reprezentované názvom, odkazom (URI) a krátkym súhrnom (angl. *snippet*). Veľkú rolu pri tom zohráva aj návyk používateľov, ktorí si len ťažko zvykajú na zmeny v zaužívaných spôsoboch vizualizácie, a tak treba každú, aj malú zmenu otestovať z hľadiska použiteľnosti (ako to bolo aj v prípade, keď Google koncom marca 2014 zrušil podčiarkovanie odkazov v zozname výsledkov[21]).

Výsledky pritom bývajú usporiadané tak, aby tie najviac relevantné pre zadaný dopyt boli v zozname čo najvyššie. Používajú sa pritom rôzne metódy, ako *PageRank, HITS,* alebo *metódy strojového učenia* (angl. súhrnne označované ako *learning-to-rank*).

Zadaný dopyt môžeme vyhľadávať naraz v rôznych doménach (správy, obrázky, videá), ktoré sa niekedy označujú ako vertikály, keďže sú navzájom ortogonálne [10]. Tu sa otvárajú otázky, ako vhodne kombinovať výsledky z rôznych vertikál (keďže tieto sú často reprezentované rôzne a sú aj ohodnocované samostatnými funkciami). Jestvujú dva základné prístupy: zmiešanie výsledkov z rôznych vertikál do jedného zoznamu, alebo prezentovanie výsledkov z každej vertikály v samostatnej časti.

Podobný problém musíme riešiť aj v prípade, ak výsledky pochádzajú z rôznych nezávislých zdrojov, napr. v prípade meta-vyhľadávača, ktorý zadaný dopyt distribuuje do rôznych samostatných vyhľadávačov a následne agreguje vrátené výsledky. Hovoríme vtedy o tzv. federatívnom vyhľadávaní (angl. *federated search*) [16].

[21] http://www.usertesting.com/blog/2014/03/21/users-have-spoken-new-google-is-better-than-old-google/

Aktuálnou témou, ktorá súvisí so zobrazovaním výsledkov a ich usporadúvaním, je *personalizácia*, resp. *zohľadnenie kontextu používateľa* pri vyhľadávaní. V prípade personalizácie sa snažíme identifikovať (modelovať) záujmy používateľa a nim prispôsobovať zobrazované výsledky [9]. Ak by používateľ napr. zadal dopyt *jaguár* a vyhľadávač by o ňom vedel, že sa zaujíma o informačné technológie a špeciálne o produkty od firmy Apple, uprednostnil by výsledky, ktoré súvisia s verziou operačného systému *OS X Jaguár*. Iný prístup predstavuje prispôsobovanie sa aktuálnemu kontextu používateľa [8] – existuje viacero typov kontextov, ako napr. *časový* (niečo iné používateľ vyhľadáva počas týždňa a iné počas víkendu, niečo iné v zime, keď si plánuje lyžovačku a iné, keď je leto a pod.). Ak by používateľ v prípade zadaného dopytu *jaguár* hľadal predtým niečo o autách, bolo by pravdepodobné, že aj teraz myslí na značku auta, a nie na operačný systém, alebo zviera, a preto by sa uprednostnili tieto výsledky.

Niektoré vyhľadávače sa nesnažia odhadnúť úmysel používateľa, ale snažia sa identifikovať všetky možné významy daného dopytu a spojiť súvisiace výsledky do zmysluplných zhlukov (na základe ich sémantickej podobnosti) [4, 19]. Pri dopyte *jaguár* by tak vyhľadávač v ideálnom prípade vrátil zhluky dokumentov o zvierati, operačnom systéme, značke áut a pod.

Súvisiacou je tiež snaha o diverzifikáciu výsledkov, t. j. aby boli výsledky, ktoré vráti vyhľadávač, čo najviac rôznorodé a ideálne pokryli čo najviac súvisiacich tém [15]. V prípade už známeho dopytu by sme tak dostali čo najviac nepodobné výsledky pokrývajúce rôzne témy (významy) zadaného dopytu, resp. rôzne výsledky v rámci jednej témy (napr. záznam na Wikipédii o tom, čo je to OS X Jaguár, stránku výrobcu Apple, používateľskú recenziu a pod.).

6.4 Vyhodnocovanie

Ak chceme zistiť, ako dobré výsledky nám vracia nami navrhnutý vyhľadávač, môžeme použiť rôzne objektívne miery. Najčastejšie používané sú v oblasti vyhľadávania informácií *presnosť* (angl. *precision*) a *úplnosť* (angl. *recall*). Presnosť nám hovorí, koľko dokumentov zo všetkých, ktoré nám vrátil vyhľadávač, je relevantných (relevantnosť je určená doménovými expertmi alebo porovnaním voči existujúcemu zlatému štandardu, napr. údajovým množinám, ktoré sú k dispozícii v rámci TREC[22]). Naproti tomu úplnosť nám hovorí, koľko zo všetkých relevantných výsledkov sa nám podarilo nájsť, t. j. koľko ich vrátil vyhľadávač. Tieto miery sú pritom vo vzájomnom vzťahu nepriamej úmery – čím je vyššia presnosť, tým nižšia je úplnosť a naopak. Cieľom je tak nájsť také nastavenie systému, ktoré bude maximalizovať obe miery (ak sa nerozhodneme niektorú z nich uprednostniť).

Často nás zaujíma presnosť len pre prvých N výsledkov – P@N. Inou častou používanou mierou je tzv. F_1 miera, ktorá predstavuje harmonický priemer presnosti a úplnosti a počíta sa takto:

$$F_1 = 2 \cdot \frac{presnosť \cdot úplnosť}{presnosť + úplnosť}$$

[22] Text Retrieval Conference: http://trec.nist.gov/

Ak chceme overiť nielen prítomnosť relevantného výsledku v zozname, ale aj správnosť jeho pozície v zozname, môžeme použiť mieru s názvom *(normalizovaný) diskontovaný kumulatívny zisk* (angl. *normalized discounted cumulative gain – NDCG*).

6.5 Ďalšie smery výskumu

Jedným z výrazných smerov výskumu je vyhľadávanie so sémantikou: aby mohli vyhľadávače rozumieť sémantike zadávaných dopytov, potrebujú poznať opis domény – entity a vzťahy medzi nimi. Existuje viacero formátov, ktoré môžu tvorcovia stránok využiť pri opise sémantiky v ich obsahu. V súčasnosti sa do popredia dostáva formát *Microdata*[23], ktorý má byť súčasťou nového (v súčasnosti ešte len vytváraného) HTML5 štandardu a *Schema.org*[24], ktorá sa vytvára ako spoločná iniciatíva firiem Google, Microsoft a Yahoo!. Metaúdaje opísané pomocou slovníka špecifikované v Schema.org sa už teraz zobrazujú vo výsledkoch vyhľadávania v podobe obohatených súhrnov (angl. *rich snippets*) [12], ako ukazuje obrázok 24.

Obrázok 24. Obohatený súhrn vo výsledku vyhľadávača Google – pod odkazom sa zobrazuje počet hviezdičiek a priemerné hodnotenie daného filmu spolu s počtom hlasov a ďalšími informáciami, ako je režisér, herci a pod.

Ešte silnejšiu (expresívnejšiu) formu sémantiky predstavujú tzv. *prepojené údaje* (angl. *Linked Data*), ktoré nám umožňujú formulovať komplikované dopyty. Ak by sme napr. chceli nájsť názvy všetkých filmov, ktoré režíroval *Tim Burton* a zároveň v nich hral *Johnny Depp*, mohli by sme v jazyku *SPARQL*[25] napísať takýto dopyt:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX movie: <http://data.linkedmdb.org/resource/movie/>

SELECT ?movieName WHERE {
  ?burton movie:director_name "Tim Burton".
  ?depp   movie:actor_name "Johnny Depp".

  ?movie  movie:director ?burton;
          movie:actor ?depp;
          rdfs:label ?movieName.
}
```

Ďalšou možnosťou, ako podporiť vyhľadávanie, je pomocou identifikácie virtuálnych komunít a ich záujmov, čo je predmetom výskumu *sociálneho vyhľadávania* [5]. Silnejšie ako sociálne väzby sú väzby, ktoré vznikajú pri spolupráci (kolaborácii) používateľov. Shah [18] definuje

[23] http://www.w3.org/TR/microdata/
[24] http://www.schema.org/
[25] http://www.w3.org/TR/rdf-sparql-query/

kolaboratívne vyhľadávanie ako proces vyhľadávania informácií, ktorý je interaktívny a vzájomne prospešný pre všetkých zúčastnených používateľov. Príkladom je spoločné vyhľadávanie – riešenie jednej úlohy viacerými používateľmi naraz (synchrónne alebo asynchrónne).

Zaujímavý smer výskumu predstavuje *prieskumné vyhľadávanie*, ktoré vo svojej práci zadefinoval Marchionini [13]. Od tradičného vyhľadávania sa líši tým, že používateľ začína s nejasnou informačnou potrebou, ktorá sa v priebehu hľadania mení, má otvorený koniec, je iteratívne a vyžaduje využitie rôznych stratégií [22, 23, 25, 26]. Úlohy prieskumného vyhľadávania sú často zložitejšie ako jednoduché dohľadanie nejakého faktu – typickými príkladom je skúmanie novej domény, keď potrebujeme zozbierať informácie v nej obsiahnuté, analyzovať ich, porovnať, agregovať a pod.

6.6 Zhrnutie

Vyhľadávanie predstavuje širokú oblasť výskumu, ktorá si nachádza praktické uplatnenie pri každodennej práci na webe. Bude zaujímavé sledovať ďalší vývoj v tejto oblasti, zrejme smerom k ešte cielenejším a presnejším výsledkom na základe znalostí záujmov používateľov, ich príslušnosti do rôznych virtuálnych skupín a aktuálneho kontextu a v neposlednom rade aj na základe pochopenia sémanticky zadávaných dopytov a k nim prislúchajúcim entitám. Môžeme tiež pozorovať väčší dôraz na používateľský zážitok a interakciu so systémom, ktorý je zastúpený v oblasti výskumu informačného hľadania a prieskumného vyhľadávania.

Literatúra

[1] Aula, A., Khan, R.M. & Guan, Z.: How does search behavior change as search becomes more difficult? In *Proc. of the 28th Int. Conf. on Human Factors in Computing Systems - CHI '10*, (2010), pp. 35–44.

[2] Bar-Yossef, Z., Kraus, N.: Context-sensitive query auto-completion. In *Proc. of the 20th Int. Conf. on World Wide Web - WWW '11*, (2011), pp. 107–116.

[3] Broder, A.: A taxonomy of web search. In *ACM SIGIR Forum*, vol. 36, no. 2, (2002), pp. 3–10.

[4] Carpineto, C. et al.: A survey of web clustering engines. *ACM Computing Surveys*, vol. 41, no. 3, (2009), pp. 1–38.

[5] Freyne, J. et al.: Collecting community wisdom: Integrating social search & social navigation. In *Proceedings of the 12th Int. Conf. on Intelligent User Interfaces - IUI '07*, (2007), pp. 52–61.

[6] Hoeber, O., Yang, X.D.: Supporting web search with visualization. In *Web-based Support Systems*, (2010), pp. 183–214.

[7] Järvelin, K., Ingwersen, P.: Information seeking research needs extension towards tasks and technology. In *Information Research*, (2004), vol. 10, no. 1, pp. 1–14.

[8] Kramár, T., Bieliková, M.: Detecting search sessions using document metadata and implicit feedback. In *WSCD 2012 Workshop on Web Search Click Data*, (2012).

[9] Kramár, T., Barla, M., Bieliková, M.: Personalizing search using socially enhanced interest model, built from the stream of user's activity. In *Journal of Web Engineering*, vol. 12, no. 1-2, (2013) pp. 65–92.

[10] Lalmas, M.: Aggregated search. In: *Advanced Topics in Information Retrieval*, (2011), pp. 109–123.

[11] Levene, M., Wheeldon, R.: Navigating the world-wide-web. In *Web Dynamics*, pp. 117–152.

[12] Haas, K. et al.: Enhanced results for web search. In *Proceedings of the 34th Int. ACM SIGIR Conference on Research and Development in Information - SIGIR '11*, (2011), pp. 725–734.

[13] Marchionini, G.: Exploratory search: from finding to understanding. *Communications of the ACM*, (2006), vol. 49, no. 4, pp. 41–46.

[14] Molnár, S., Móro R., Bieliková, M.: Trending words in digital library for term cloud-based navigation. In *SMAP '13: Proceedings of the 8th International Workshop on Semantic and Social Media Adaptation and Personalization*, (2013), pp. 53–58.

[15] Panigrahi, D. et al.: Online selection of diverse results. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining - WSDM '12*, (2012), pp. 263–272.

[16] Ponnuswami, A.K. et al.: On composition of a federated web search result page: using online users to provide pairwise preference for heterogeneous verticals. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining - WSDM '11*, (2011), pp. 715–724.

[17] Rasiwasia, N. & Vasconcelos, N.: Image retrieval using query by contextual example. In *Proceedings of the 1st ACM International Conference on Multimedia Inf. Retrieval - MIR '08*, (2008), pp. 164–171.

[18] Shah, C.: Collaborative Information Seeking in Context. In *Collaborative Information Seeking*, (2012), pp. 25–39.

[19] Turetken, O., Sharda, R.: Clustering-based visual interfaces for presentation of web search results: An empirical investigation. In *Information Systems Frontiers*, vol. 7, no. 3, (2005), pp. 273–297.

[20] Tvarožek, M., Bieliková, M.: Generating exploratory search interfaces for the semantic web. In *Human-Computer Interaction, IFIP Advances in Information and Communication Technology*, (2010), pp. 175–186.

[21] White, R.W., Morris, D.: Investigating the querying and browsing behavior of advanced search engine users. In *Proceedings of the 30th Annual International ACM SIGIR Conf. on Research and Development in Information Rretrieval - SIGIR '07*, (2007), pp. 255–262.

[22] White, R.W., Roth, R.A.: *Exploratory search: beyond the query-response paradigm*, Morgan & Claypool, (2009).

[23] Wilson, M.L. et al.: From keyword search to exploration: Designing future search interfaces for the web. In *Foundations and Trends in Web Science*, vol. 2, no. 1, (2010), pp.1–97.

[24] Wilson, T.: Models in information behaviour research. In *Journal of documentation*, vol. 55, no. 3, (1999), pp. 249–270.

[25] Návrat, P.: Cognitive traveling in digital space: from keyword search through exploratory information seeking. In *Central European Journal of Computer Science*, vol. 2, issue 3, (2012), pp. 170-182.

[26] Tvarožek, M.: Exploratory Search in the Adaptive Social Semantic Web. Information Sciences and Technologies. In *Bulletin of the ACM Slovakia*, vol. 3, no. 1, (2011), pp. 42-51.

7 Ako funguje webový vyhľadávač

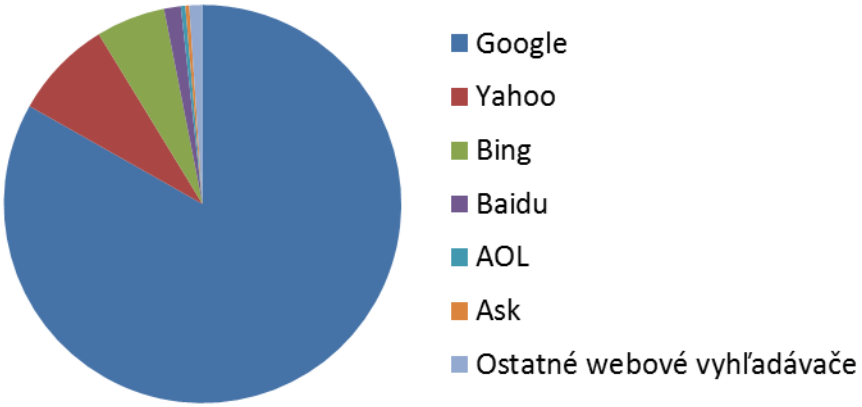
Webový vyhľadávač ako napr. Google, či Bing sa postupne stáva neodmysliteľnou súčasťou nášho života. S rastúcim počtom webových stránok vznikla potreba efektívneho vyhľadávania informácií na webe, v dôsledku čoho sa vyvinuli viaceré webové vyhľadávače a v súčasnosti predstavujú jednu z najpoprednejších oblastí informatiky. Webové vyhľadávače v sebe ukrývajú aj množstvo výskumných problémov, ktoré spájajú dokopy viacero vedných disciplín od lingvistiky a kognitívnej vedy až po matematiku, fyziku a informatiku. To robí túto oblasť ešte viac zaujímavejšou nielen pre výskumníkov, ale aj pre komerčné spoločnosti vyvíjajúce webové vyhľadávače, ktoré medzi sebou súperia, aby získali čo najviac používateľov.

Pojem vyhľadávač pochádza z oblasti vyhľadávania informácií. V tejto kapitole sa zameriame na jeden z najrozšírenejších typov – webový vyhľadávač. Popularita webových vyhľadávačov odzrkadľuje potrebu efektívneho vyhľadávania informácií na webe, ktorý neustále rastie a v súčasnosti predstavuje jeden z najproduktívnejších zdrojov údajov na svete. Webový vyhľadávač možno rozdeliť na tri funkčné celky:

- preliezač webu
- indexovač slov webových dokumentov
- vyhľadávací mechanizmus, ktorý analyzuje prepojenia medzi webovými stránkami a umožňuje používateľom vyhľadávať dokumenty pomocou indexovaných slov a ich kombinácií.

7.1 História

Prvý nástroj na vyhľadávanie na internete vznikol v roku 1990 a nazýval sa Archie. Vedel sťahovať zoznam súborov na verejných anonymných FTP serveroch, ktoré indexoval v jednoduchej databáze názvov súborov. Archie však ešte neindexoval obsah súborov, keďže vtedajší malý počet súborov bolo možné jednoducho prejsť celkom rýchlo aj ručne. V roku 1991 vznikol Gopher, ktorý už dokázal indexovať aj obsah textových dokumentov. V roku 1998 vznikol Google, ktorý už poznáme i dnes a predstavuje dominantu medzi webovými vyhľadávačmi (pozri Obrázok 25). V roku 1998 vznikol aj vyhľadávač MSN Search od firmy Microsoft, ktorý od roku 2009 poznáme pod názvom Bing a od toho istého roku tvorí aj jadro vyhľadávača Yahoo!. V roku 2000 vznikol čínsky webový vyhľadávač Baidu, ktorý sa prednostne zameriava na vyhľadávanie webových stránok v čínštine.



Obrázok 25. Celosvetová používanosť webových vyhľadávačov[26].

7.2 Preliezač webu

Preliezač webu je internetový bot, ktorý systematicky prehliada web. Začína s úvodným zoznamom URL adries, ktoré chce navštíviť. Adresy v tonto zozname sa nazývajú aj semiačka preliezača, pretože postupným prehliadaním webových dokumentov sa úvodný zoznam URL adries „rozrastie“ na zoznam všetkých URL adries, ktoré preliezač vôbec navštívi. Preliezač postupne sťahuje webové dokumenty z URL adries v zozname a z každého dokumentu extrahuje všetky (hyper)prepojenia, ktoré ďalej pridáva do zoznamu. Z tohto procesného pohľadu sa preto priebežný zoznam URL adries označuje ako preliezací front. URL adresy v preliezacom fronte sa rekurzívne navštevujú podľa niekoľkých politík.

7.2.1 Preliezacie politiky

Existuje niekoľko preliezacích politík, ktoré by mal každý preliezač pri preliezaní webu dodržiavať:

[26] http://www.netmarketshare.com/search-engine-market-share.aspx?qprid=4&qpcustomd=0

- politika výberu
- politika znovunavštevovania
- politika zdvorilosti
- politika paralelizácie.

Politika výberu stanovuje, ktoré stránky má preliezač sťahovať. Podľa prieskumu z roku 2005 [4] dokonca aj rozsiahle preliezače zindexujú iba 40-70% celého indexovateľného webu. Preto je veľmi dôležité, aby prelezená časť webu obsahovala tie najrelevantnejšie stránky a nie len nejakú náhodnú vzorku webu. Niektoré stránky môžu tiež obsahovať tzv. preliezacie pasce, ktoré predstavujú nekonečný zoznam dynamicky generovaných URL adries. Preto je dôležité, aby preliezač rešpektoval robotický protokol, ktorý uvádza, akú časť sídla je možné preliezať.

Politika znovunavštevovania stanovuje, ako často kontrolovať zmeny na stránkach. Preliezače používajú viaceré metriky, podľa ktorých sa rozhodujú, kedy znovu navštíviť nejakú webovú stránku. Najčastejšie používané metriky sú čerstvosť a vek [7]. Čerstvosť je jednoduchá binárna metrika, ktorá indikuje, či sa lokálna kópia stránky zhoduje s originálom, alebo nie. Čerstvosť webovej stránky *s* v čase *t* je definovaná ako:

$$F_p(t) = \begin{cases} 1 & ak\ sa\ webová\ stránka\ s\ zhoduje\ s\ lokálnou\ kópiou\ v\ čase\ t \\ 0 & v\ opačnom\ prípade \end{cases} \quad (1)$$

Vek je metrika, ktorá indikuje ako stará je lokálna kópia. Vek webovej stránky *s* v čase *t* je definovaný ako:

$$A_p(t) = \begin{cases} 0 & ak\ je\ webová\ stránka\ s\ v\ čase\ t\ nezmenená \\ t - čas\ zmeny\ stráky\ s & v\ opačnom\ prípade \end{cases} \quad (2)$$

Politika zdvorilosti stanovuje, ako často navštevovať rovnaký webový server. Jej účelom je zabrániť preťažovaniu webových serverov pri preliezaní webu, keďže preliezače dokážu prehliadať webové stránky oveľa rýchlejšie ako ľudia, ktorí ich prehliadajú ručne. Čiastočným riešením je opäť robotický protokol, väčšina preliezačov si však stanuje časový interval medzi nasledovnými stiahnutiami. Veľkosť tohto intervalu je zvyčajne pár sekúnd [1, 8]. Treba však poznamenať, že v praxi neexistuje univerzálne optimálne nastavenie [3].

Politika paralelizácie je určená pre paralelné preliezače, ktoré majú niekoľko paralelných procesov. Stanovuje, ako koordinovať distribuované webové preliezače s cieľom maximalizovať rýchlosť preliezania webu, minimalizovať režijné náklady na paralelizáciu a zabrániť viacnásobnému sťahovaniu rovnakých stránok.

7.2.2 Problémy pri preliezaní webu

Pri preliezaní webu môžeme natrafiť na viaceré problémy:

- Ktoré URL adresy použiť na inicializáciu preliezania?
- Ktoré URL adresy sa majú navštíviť skôr?
- Ako udržiavať aktuálny index?
- Ako preliezať čo najviac stránok?

- Ako nepreliezať duplicitný obsah?
- Ako sa vyhnúť spamu a nepreliezať ho?
- Ako sa vyhnúť pasciam na preliezače?
- Ako preliezť aj hlboký web?

Aj vzhľadom na rýchly rast webu sú mnohé z týchto problémov doposiaľ aktuálne a vyžadujú aktívny výskum. Existujú však viaceré riešenia, ktoré sa snažia s týmito problémami vysporiadať. Na inicializáciu preliezania sa zvyknú používať webové adresáre ako Yahoo, či ODP. Pri rozhodovaní, ktoré stránky sa majú navštíviť skôr, sa zvykne použiť niekoľko alternatív:

- prehliadanie do šírky (tzv. FIFO princíp – angl. first-in-first-out)
- stránky, ktoré sa menia častejšie
- populárne stránky (napr. podľa hodnoty PageRank).

Pasce na preliezače sú nekonečné grafy dynamicky prepojených stránok ako napr. kalendár. Takýmto pasciam sa preliezač dokáže vyhnúť najmä rešpektovaním spomínaného robotického protokolu.

S cieľom preliezania hlbokého webu sa vytvárajú inteligentné preliezače, ktoré nie sú len obyčajné sťahovače webových stránok, ale vedia stránku úplne interpretovať rovnako ako webový prehliadač a dokážu interagovať aj s aktívnymi prvkami (rozličné technológie ako Adobe Flash), či vypĺňať formuláre na stránke, ktoré sprístupňujú inak skrytú časť hlbokého webu.

7.3 Indexovanie webových stránok

Cieľom indexovania webových stránok je umožniť rýchle a presné vyhľadávanie informácií. Pri preliezaní sa obsah indexuje a odkazy sa ukladajú do databázy na ďalšie spracovanie. Problém indexovania spája v sebe viaceré disciplíny ako sú lingvistika či matematika. Pri indexovaní musíme riešiť niekoľko problémov:

- spájanie údajov
- ako ukladať údaje
- veľkosť indexu
- rýchlosť vyhľadávania
- správa indexu
- tolerancia chybovosti.

Existuje viacero indexovacích štruktúr, ktoré riešia uvedené problémy v rozličnej miere:

- príponový strom
- invertovaný index
- citačný index
- N-gramový index
- matica výskytu dokumentov a výrazov.

Najpopulárnejší typ indexu nielen pre webové vyhľadávače je invertovaný index [12]. Je to údajová štruktúra, ktorá zaznamenáva, na ktorých stránkach sa vyskytovali ktoré tokeny.

Podobne sa používajú viaceré metriky na meranie popularity slova pre daný dokument, či dopyt. Jednou z najpopulárnejších takýchto metrík je metrika tf-idf [11] (angl. term frequency – inverse document frequency, v preklade: frekvencia výrazu – inverzná frekvencia v dokumente). Myšlienkou tohto prístupu je znevýhodniť vysoko frekventované výrazy, ktoré majú nízku rozlišovaciu schopnosť relevancie dokumentu (rovnica 3). tf-idf pozostáva z dvoch častí:

- tf(v, d) – frekvencia výrazu v dokumente, t.j. počet výskytov výrazu v v d
- idf(v) – prevrátená hodnota podielu dokumentov, v ktorých sa vyskytuje v, na všetkých dokumentoch, ktorá je zároveň ekvivalentná zápornému logaritmu pravdepodobnosti výskytu výrazu v dokumente (rovnica 4)

$$TFIDF(v,d) = TF(v,d) \cdot IDF(v) \tag{3}$$

$$IDF(v) = log\frac{|D|}{|\forall d \in D : v \in d|} = -log\frac{|\forall d \in D : v \in d|}{|D|} = -logP(v) \tag{4}$$

kde v je výraz, d dokument a $|D|$ reprezentuje celkový počet dokumentov.

7.4 Analýza prepojení

Okrem obsahu vieme využiť aj prepojenia medzi stránkami na zlepšenie usporiadania výsledkov vyhľadávača. Ukazuje sa, že využitím informácie o prepojeniach medzi webovými stránkami, teda analýzou grafu prepojení, vieme nájsť podstatne lepšie výsledky ako len na základe samotného obsahu stránok.

7.4.1 Čo vyjadruje prepojenie?

Existencia prepojenia medzi dvoma stránkami môže mať rôzny význam. Nejednoznačnosť významu prepojenia preto spôsobila aj vznik viacerých rôznych algoritmov na výpočet relevancie a usporiadanie webových stránok práve na základe prepojení medzi nimi. Najbežnejšie významy prepojenia medzi stránkami A a B (pozri Obrázok 26) sú:

- A odporúča B
- A obzvlášť neodporúča B
- B odkazuje na A ako na autoritu
- A a B sú o tej istej veci (lokalita tém)

Obrázok 26. Prepojenie medzi stránkami - stránka A sa odkazuje na stránku B.

7.4.2 História

Myšlienka formulovania problému analýzy prepojení ako problému vlastného čísla (angl. eigenvalue) bola pravdepodobne prvýkrát navrhnutá v roku 1976 Gabrielom Pinskim a Francisom Narinom, ktorí pracovali na scientometrickom usporiadaní vedeckých článkov. PageRank vymysleli vtedajší študenti Stanfordovej univerzity Larry Page a Sergey Brin v roku 1996 v rámci

výskumného projektu zameraného na vytvorenie nového vyhľadávača. V roku 1998 publikovali prvý článok o algoritme PageRank a prvotnom prototype vyhľadávača Google. Krátko potom (1998) založili spoločnosť Google, Inc., ktorá je dnes jednou z najúspešnejších počítačových firiem na svete. V roku 1996 Robin Li vytvoril RankDex algoritmus na usporiadanie výsledkov vyhľadávača, ktorý v roku 1999 patentovali. Tento algoritmus neskôr vytvoril základ vyhľadávača spoločnosti Baidu, ktorú založili 1. januára 2000. V roku 1998 Jon Kleinberg publikoval svoj článok o HITS (Hyperlink-Induced Topic Search).

7.4.3 PageRank a model náhodného surfistu

Neodškriepiteľne najúspešnejší algoritmus na usporiadanie webových stránok je PageRank [3], ktorý umožnil vznik v súčasnosti najpopulárnejšieho vyhľadávača na svete, známeho ako Google. PageRank nesie meno jedného zo svojich vynálezcov – Larryho Pagea.

Na základe prepojení počíta PageRank dôležitosť webových stránok. Základnou myšlienkou algoritmu je spočítanie počtu a kvality prepojení, ktoré odkazujú na nejakú stránku. PageRank má niekoľko predpokladov:

- na dôležitejšie webové stránky ostatné stránky odkazujú častejšie ako na menej dôležité webové stránky
- prepojenia z dôležitejších webových stránok sú významnejšie ako prepojenia z menej dôležitých webových stránok
- čím viac prepojení má webová stránka, tým majú menšiu váhu.

Pre lepšie pochopenie princípu, na ktorom funguje PageRank, sa využíva model náhodného surfistu. Surfista začína na náhodnej stránke a náhodne si vyberá prepojenie, cez ktoré prejde na ďalšiu stránku. PageRank stránky vyjadruje pravdepodobnosť, s akou náhodný surfista skončí na tejto stránke v ľubovoľnom čase (po prejdení akéhokoľvek počtu prepojení). Samotný model náhodného surfistu však nerieši problém pozičných prepadlísk. Medzi takéto pozičné prepadliská patrí aj tzv. obšmietajúci sa uzol alebo cyklus, z ktorého sa nedá dostať. Vo všeobecnosti sú pozičné prepadliská všetky také silne súvislé komponenty grafu prepojených webových stránok, z ktorých neexistuje cesta do žiadneho iného silne súvislého komponentu. Pre náhodného surfistu teda predstavujú akúsi slepú uličku, z ktorej niet úniku. Preto algoritmus PageRank rieši tento problém náhodnou teleportáciou na ľubovoľný iný uzol. Ku každému rozhodnutiu náhodného surfistu sa pridá pravdepodobnosť teleportácie α:

- šanca α, že sa surfista začne nudiť a preskočí na niektorý iný uzol grafu
- šanca (1-α), že surfista si vyberie jedno z dostupných prepojení.

Formálne vypočítame hodnotu PageRank stránky P_i podľa rovnice 5, kde $|P|$ je počet všetkých webových stránok, B_{Pi} je množina všetkých stránok, ktoré sa odkazujú na stránku P_i a $|P_j|$ je počet všetkých prepojení na stránke P_j. Typické nastavenie pravdepodobnosti teleportácie α je 0.15.

$$PR(P_i) = \frac{\propto}{|P|} + (1-\propto) \cdot \sum_{P_j \in B_{P_i}} \frac{PR(P_j)}{|P_j|} \qquad (5)$$

Podľa uvedenej rovnice vieme pre každú webovú stránku vypočítať jej hodnotu PageRank. Je však potrebných viacero iterácií, keďže na začiatku nepoznáme hodnotu PageRank pre žiadnu stránku. V praxi však stačí niekoľko desiatok iterácií, kým hodnoty PageRank neskonvergujú. Navyše vieme uplatniť aj maticové násobenie prepisom rovnice 5 do vektorovo-maticovej reprezentácie (rovnica 6).

$$R_{i+1} = TR_i \tag{6}$$

V uvedenej rovnici máme hodnoty PageRank reprezentované ako jeden vektor R_i a matica T predstavuje prechodovú maticu, ktorej vynásobením s vektorom R_i dostaneme nový vektor R_{i+1}, ktorý reprezentuje hodnoty PageRank.

Problémy

Základným problémom algoritmu PageRank je, že „bohatí sa stávajú bohatšími“. Nové stránky s málo vstupnými prepojeniami len ťažko súperia so staršími početne odkazovanými s vysokou hodnotou PageRank.

V minulosti bol PageRank celkom jednoducho manipulovateľný. Presmerovanie formou HTTP 302 odpovede alebo metatagom „Refresh“ (angl. obnoviť) spôsobovalo, že stránka s presmerovaním nadobudla hodnotu PageRank (PR) cieľovej stránky. Týmto spôsobom dokázala stránka s PR 0 so žiadnymi prepojeniami na ňu nadobudnúť PR 10 presmerovaním na domovskú stránku Google.

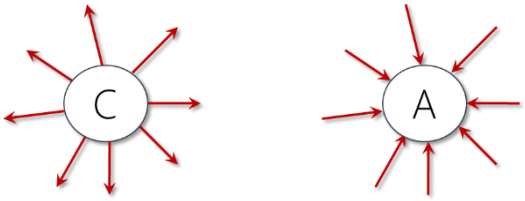
Ďalším problémom, ktorý vznikol, bolo kupčenie s odkazmi, ktoré majú vysokú hodnotu PR[27]. Keďže odkazy, ktoré majú vyššiu hodnotu PR sú vzácnejšie, zvyknú byť preto aj drahšie. Google však verejne varoval správcov webov, že ak ich objaví kupčiť s prepojeniami s cieľom zmanipulovať PR, tak všetky ich odkazy znehodnotí a nebude ich uvažovať pri počítaní PR ostatných stránok.

7.4.4 HITS

Populárna alternatíva algoritmu PageRank je algoritmus HITS - prepojeniami indukované vyhľadávanie tém [9]. Na rozdiel od algoritmu PageRank, HITS rozlišuje dva typy uzlov – centrá a autority (viď Obrázok 27). Myšlienka algoritmu stavia na dvoch hypotézach:

- Dobré centrá odkazujú na dobré autority
- Na dobré autority sa odkazujú dobré centrá.

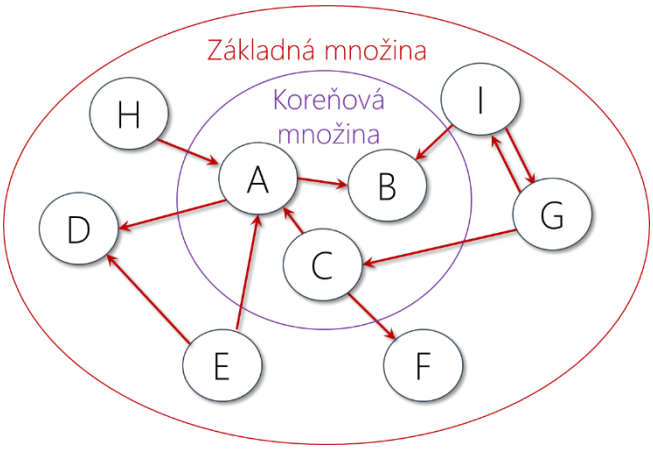
[27] "How to report paid links". mattcutts.com/blog.



Obrázok 27. Dva typy uzlov v algoritme HITS. C označuje centrum a A označuje autoritu.

Algoritmus HITS postupuje v týchto krokoch:

1. Zober stránky najrelevantnejšie pre daný vyhľadávací dopyt – koreňová množina (pozri Obrázok 28)
2. Zober stránky, ktoré sú prepojené s koreňovou množinou – základná množina (pozri Obrázok 28)
3. Počítaj iteratívne hodnoty autorít a centier nad všetkými uzlmi v podgrafe
4. Na konci má každý uzol hodnotu autority a centra



Obrázok 28. Koreňová a základná množina pri výpočte algoritmu HITS.

Výpočet hodnoty autority je uvedený v rovnici 7 a výpočet hodnoty centra v rovnici 8.

$$A(p) = \sum_{q:e_{qp} \in E} C(q) \quad (7)$$

$$C(p) = \sum_{q:e_{pq} \in E} A(q) \quad (8)$$

E predstavuje množinu všetkých orientovaných hrán a e_{pq} predstavuje prepojenie zo stránky p na stránku q. Podobne ako v algoritme PageRank, aj algoritmus HITS vyžaduje výpočet vo viacerých iteráciách, až kým hodnoty centier a autorít neskonvergujú.

Problémy

Jedným z problémov oproti algoritmu PageRank je, že algoritmus HITS nikdy nepoužili vo väčšom rozsahu, pretože IBM naň vlastní patent. Navyše algoritmus HITS je závislý na dopyte, takže sa vykonáva až po zadaní dopytu, nie počas indexovania, čo má značný dopad na jeho časovú efektívnosť pri spracovaní dopytu.

7.4.5 Spam v prepojeniach

Postupne s rastúcim rozsahom webu sa začal objavovať aj spam v prepojeniach. Mnohé odkazy neboli vôbec relevantné, čo kazilo kvalitu vypočítaných hodnôt pri analýze prepojení. Vznikali celé farmy prepojení, ktorých účelom bolo zmanipulovať kvalitu výsledkov vyhľadávania. Začali sa praktizovať výmeny recipročných prepojení. Na kvalitu výsledkov tiež negatívne vplývalo uvádzanie odkazov na blogoch a fórach. V roku 2005 Google zaviedol používanie atribútu „no follow“ (angl. nenasleduj) pri linkách, ktoré sa majú ignorovať pri analýze prepojení, s cieľom bojovať proti takémuto tzv. „spamdexovaniu“ [6].

S cieľom bojovať proti spamu v prepojeniach vzniklo viacero algoritmov. V roku 2004 vznikol TrustRank [5] – technika analýzy prepojení, ktorá preferuje teleportáciu na množinu dôveryhodných stránok. V roku 2005 vznikol SpamRank [2], ktorý penalizuje stránky porušujúce zákon rozdelenia sily. V roku 2006 vznikol Anti-TrustRank [10], ktorý dáva vysokú hodnotu známym spamovým stránkam a túto informáciu propaguje použitím algoritmu PageRank.

7.1 Zhrnutie

Webové vyhľadávače reprezentujú jednu z najdôležitejších súčastí webu, najmä vzhľadom na neustále sa zväčšujúci počet webových stránok na webe. Tento prudký rast má dopad aj na množstvo ďalších oblastí. Takmer každým dňom sa objavujú doposiaľ nepoznané problémy, ktorými sa rozširuje oblasť vyhľadávania na webe čoraz viac a viac. Preto ani zďaleka nemôžeme považovať túto kapitolu za vyčerpávajúci opis webových vyhľadávačov, ale skôr za úvod do problematiky, ktorý čitateľa oboznámi so základnými konceptami.

Referencie

[1] Baeza-Yates, R., Castillo, C.: Balancing volume, quality and freshness in Web crawling. In *Soft Computing Systems – Design, Management and Applications*, pages 565–572, Santiago, Chile. IOS Press Amsterdam, 2002.

[2] Benczur, Andras A., et al.: SpamRank–Fully Automatic Link Spam Detection Work in progress. *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web*. 2005.

[3] Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998.

[4] Gulli; A., Signorini, A.: The indexable web is more than 11.5 billion pages. *Special interest tracks and posters of the 14th international conference on World Wide Web*. ACM Press. pp. 902–903, 2005.

[5] Gyöngyi, Z.; Garcia-Molina, H., Pedersen, J.: Combating Web Spam with TrustRank. *Proceedings of the International Conference on Very Large Data Bases* 30: 576, 2004.

[6] Gyöngyi, Z.; Garcia-Molina, H.: Web spam taxonomy, *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb), 2005 in The 14th International World Wide Web Conference (WWW 2005)*, 2005, Nippon Convention Center (Makuhari Messe), Chiba, Japan., New York, NY: ACM Press, ISBN 1-59593-046-9.

[7] Cho, J.; Garcia-Molina, H.: Synchronizing a database to improve freshness. *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. Dallas, Texas, United States: ACM. pp. 117–128. ISBN 1-58113-217-4, 2000.

[8] Cho, J., Garcia-Molina, H.: Effective page refresh policies for web crawlers. *ACM Transactions on Database Systems*, 28(4), 2003.

[9] Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. In *Journal ACM* 46, vol. 5, (1999), pp. 604-632.

[10] Krishnan, V., Raj, R.: Web Spam Detection with Anti-Trust Rank. (2006). Dostupné na: http://infolab.stanford.edu/~kvijay/krishnan-raj-airweb06.pdf

[11] Salton, G., Buckley, Ch.: Term-weighting approaches in automatic text retrieval. In *Information Processing and Management: an International Journal*, vol. 24, issue 5, (1988), pp. 513-523.

[12] Zobel, J., Moffat, A., Ramamohanarao, K.: Inverted files versus signature files for text indexing. In *ACM Transactions on Database Systems*, vol. 23, issue 4, (1998), pp. 453-490.

8 Preliezač webu v jazyku python

Jazyk python je jedným z mladých, moderných jazykov, ktoré robia vytváranie webových aplikácií ešte jednoduchším. Tento jazyk sa však od väčšiny dnes používaných jazykov odlišuje v používaní odsadení v kóde na označenie blokov. Dobre sa zapísal u komunity webových vývojárov a existuje preň v tejto oblasti veľké množstvo knižníc a rozšírení. Práve preto jazyk prináša veľmi dobré podhubie pre vytvorenie webového preliezača. Ale aj v tomto jazyku je nutné dávať pozor na nástrahy webu a vytvárať preliezač, ktorý dokáže skonštruovať správnu adresu z relatívnej cesty. Okrem toho je nutné dbať na identifikovanie duplicitných adries pomocou ich normalizácie. Bez takýchto opatrení by preliezač mohol veľmi rýchlo naraziť v dynamickom svete webu na problémy.

Programovací jazyk python je mladý a moderný jazyk. Tento programovací jazyk vznikol ako skriptovací jazyk pre operačný systém Amoeba OS. Cieľom tohto operačného systému je docieliť, aby sa celá sieť počítačov tvárila pre používateľa ako jeden počítač. Tento jazyk vytvoril holandský programátor Guido van Rossum. História toto programovacieho jazyka sa začala písať v decembri 1989 [7]. Guido o začiatkoch jazyka python povedal toto:

„Pred šiestimi rokmi, v decembri 1989, som hľadal zábavný programátorský projekt, ktorý by ma zamestnal cez týždeň počas Vianoc. Moja kancelária bola zavretá, ale mal som domáci počítač a nič iného na práci. Rozhodol som sa, že napíšem interprét pre nový skriptovací jazyk, o ktorom som už skôr premýšľal: nasledovník jazyka ABC, ktorý by zaujal programátorov v Unixe/C. Ako pracovný názov som zvolil python, lebo som bol v nevážnej nálade (a tiež som veľkým fanúšikom Monty Pythonovho Lietajúceho cirkusu).“[1]

Neskôr v roku 1999 Guido definoval hlavné ciele jazyka python:

- ľahký a intuitívny jazyk, ktorý je zároveň dostatočne mocný, aby obstál medzi hlavnými konkurentmi

- otvorený kód, takže sa môže každý zapojiť do jeho vývoja
- kód, ktorý je zrozumiteľný ako bežná angličtina
- vhodný pre bežné každodenné úlohy, umožňujúci vývoj v krátkom čase.

8.1 Programovací jazyk python

Jazyk python je multiparadigmovým jazykom, vďaka čomu je na programátorovi, akým štýlom bude samotný program písať. Jazyk podporuje objektovo-orientované programovanie, aspektovo-orientované programovanie a funkcionálne programovanie.
Podporuje sa aj dynamické typovanie, vďaka čomu programátor nemusí definovať konkrétny typ premennej, ako je to v iných jazykoch ako C, java alebo C#. Okrem toho od verzie 2 obsahuje čističku pamäti (angl. garbage collector). Základná vlastnosť jazyka python je aj to, že obsahuje neskoré viazanie, ktoré viaže mená premenných a metód až počas vykonávania programu [1].

8.1.1 Syntax

Syntax používaná v jazyku python je zameraná na to, aby bol kód veľmi dobre čitateľný. python používa sadu kľúčových slov v angličtine. Jeho základnou odlišnosťou od ostatných bežných jazykov je to, že namiesto značiek pre bloky používa odsadenia. Tento spôsob písania kódu prebral python od jeho predchodcu jazyka ABC. Na nasledujúcej ukážke môžeme vidieť rozdiel medzi kódom v jazyku python a kódom v jazyku C:

```c
void test(int count)
{
    if(count == 0)
    {
        return;
    }
    else if(count > 10)
    {
        count = count-2;
    } else {
        count = count-1;
    }
    test(count);
}
```

```python
def test(count):
    if(count == 0):
        return
    elif(count > 10):
        count = count-2
    else:
        count = count-1
    test(count)
```

Na ľavej strane sa nachádza kód v jazyku C a na pravo kód v jazyku python. Na prvý pohľad je jasne vidieť, že zápis v jazyku python je značne odľahčený, čo je najmä vďaka faktu, že odsadenia sú syntakticky významné. V jazyku python teda nie je nutné písať znaky pre začiatok a koniec bloku.
V našom kóde je napríklad zachytená podmienka, ktorá podľa hodnoty premennej *count*, vykoná akciu. Ako vidíme, v podmienke sa mení premenná *count*. Podmienka následne končí na riadku,

Prekvapenia v jazyku python

Tvorcovia jazyka schovali niekoľko prekvapení do jazyka.

Napríklad v prípade, ak zadáte príkaz `from ___future___ import brackets`, kód vyhodí výnimku: `SyntaxError: not a chance`.

Ďalej, ak zadáte príkaz `import this`, python vypíše celú jeho filozofiu.

Pre vypísanie textu *„Hello world!“* stačí zadať `import ___hello___` [9].

kde je volanie funkcie test, ktoré je odsadené rovnako, ako samotná podmienka. Tento zápis okrem iného núti programátora dodržiavať dobré zvyky pri písaní kódu.

Jazyk obsahuje všetky štandardné údajové typy ako text (str), čísla (int, float, complex), logické hodnoty (bool). Okrem toho sú veľmi dobre podporované rôzne kolekcie:

- zoznam (angl. list) – môže obsahovať rôzne prvky, dá sa v ňom pohybovať, pridávať a uberať prvky
- n-tica (angl. tuple) – je to nemeniteľný zoznam prvkov. Môže sa využívať ako spôsob veľkého množstva parametrov medzi funkciami a podobne
- sada (angl. set) – môže obsahovať rôzne údajové typy, ktoré musia byť ale hashovateľné. Nemôže obsahovať duplicity. Môže byť aj statický aj meniteľný
- slovník (angl. dict) – obsahuje dvojice kľúč-hodnota. Kľúče nemôžu byť duplicitné. Je ho možné prechádzať a pracovať s prvkami.

8.1.2 Implementácie

Jazyk python má niekoľko implementácií. Základná implementácia je CPython, napísaná v jazyku C. Je dostupná pre operačný systém Windows a často sa využíva v rôznych zariadeniach založených na jadre Unix. Okrem toho existuje PyPy interprét, ktorý sa zakladá na verzii 2.7. Ide o veľmi rýchlu a výkonnú implementáciu jazyka. Obsahuje aj podporu viacjadrových procesorov [4].

Ďalšími interpretáciami sú:

- Jyton – kompiluje sa na javovský binárny kód, ktorý potom môže bežať na *Java Virtual Machine*
- IronPython – podobný ako *Jython*, ale táto interpretácia kompiluje kód do *.Net*
- Pyjamas – kompiluje python kód do javascriptu,
- PyS60 – vytvorila spoločnosť *Nokia* pre telefón S60 s operačným systémom *Symbian*.

8.2 Preliezač webu

K pojmu preliezač webu sme sa dostali aj v jednej z predchádzajucich kapitol. Takýto pojem sa často používa pre programy na robotické prechádzanie webových stránok. Tieto programy sa väčšinou používajú na účely získavania údajov pre následnú indexáciu a využitie vo vyhľadávačoch.

Preliezač dostáva ako vstup sadu webových stránok, ktoré slúžia na prvé odrazenie sa vo vyhľadávaní. Následne si preliezač už vystačí sám a nové stránky objaví svojimi silami. Ako prvé preliezač navštevuje prvú adresu, ktorú dostane a stiahne jej obsah. Z tohto obsahu extrahuje všetky ďalšie odkazy, ktoré si zaraďuje do zoznamu na ďalšie navštívenie. Takýmto spôsobom potom preliezač postupne objavuje nové a nové stránky. Schému takéhoto preliezača môžeme vidieť aj na obrázku 29 [12].

Obrázok 29. Schéma preliezača. Preliezanie sa začína úvodným krokom cez zoznam URL. Následne sa tieto URL prechádzajú, získavajú sa nové adresy a zaraďujú sa do repozitárov.

Priestor webu je však nevyspytateľný a preliezač si v ňom musí dávať pozor na niekoľko vecí:

- stratégia výberu stránok
- spôsob ochrany voči zacykleniu
- spôsob paralelizácie
- opätovné navštevovanie stránok.

8.3 Preliezač v jazyku python

V ďalšej časti tejto kapitoly ukážeme spôsob vytvorenia webového preliezača v jazyku python. Počas vytvárania webového preliezača budeme upozorňovať čitateľa na bežné chyby a problémy, na ktoré si pri takomto type softvéru treba dať pozor.

8.3.1 Získanie obsahu webovej stránky

Náš prehliadač začneme vyvíjať od samotného jadra, ktorým je sťahovanie stránok. V pythone na tento účel slúži knižnica urllib.request [9], ktorú vo verzii 3 jazyka úplne prepracovali.

```
import urllib.request

def get_page_content(pageName):
    request = urllib.request.Request(pageName)
    request.add_header('User-Agent', 'FIIT preliezač preliezac.fi.it')
    response = urllib.request.urlopen(request)
    html = response.read()
    return str(html)
```

Pomocou tejto knižnice vieme vytvoriť požiadavku na stiahnutie stránky s konkrétnou adresou. V našom kóde následne pridávame identifikáciu preliezača pomocou hlavičkového parametra

User-Agent. Tento bežne slúži na identifikáciu prehliadača. V prípade vytvárania webového preliezača sa odporúča použiť tento parameter pre informácie o vašom preliezači.

Správcovia www stránok, ktoré navštevujete, nebudú vďaka nemu zmätení pri čítaní záznamu činnosti. Odporúča sa v tomto parametri uviesť aj kontakt na vývojový tím preliezača. Tento kontakt potom administrátori môžu využiť v prípade, keď chcú kontaktovať vývojárov preliezača. Náš kód následne len otvorí webovú stránku a získa jej obsah. Tento následne vracia ako výstup z danej funkcie.

Šetrenie zdrojov

Náš sťahovač by bolo následne ešte možné obmedziť na sťahovanie iba stránok typu HTML príkazom:

```
content_type = response.info().get('Content-Type')
If(content_type != 'text/html')
   raise UnsupportedFileTypeError()
```

Volanie tohto kódu by sme vložili ešte pred stiahnutie samotnej stránky. Táto séria príkazov totiž sťahuje iba hlavičku dokumentu a z nej zisťuje typ obsahu, ktorý sa nám pošle. V prípade, ak nejde o spomínané HTML, konkrétnu stránku nesťahujeme. Takýmto spôsobom šetríme čas a prostriedky nášho preliezača a aj stránky, z ktorej obsah sťahujeme.

8.3.2 Hľadanie odkazov v stránkach

Keď už máme konkrétnu stránku stiahnutú zo servera, je nutné danú stránku prehľadať a nájsť všetky odkazy. Odkazy sa v HTML kóde značia pomocou značky **a**. Na to, aby sme takúto značku v kóde našli, je niekoľko možností:

- klasické vyhľadávanie v reťazcoch,
- vyhľadávanie pomocou regulárnych výrazov,
- použitie externej knižnice na prácu s HTML súbormi.

Každý z prístupov má svoje výhody a nevýhody. Azda najhorším je klasické vyhľadávanie v texte. Toto je v prvom rade veľmi pomalé, ale aj náročné a nespoľahlivé s ohľadom na spôsob, akým sa v dnešnom webe tvoria webové stránky.

Vyhľadávanie pomocou regulárnych výrazov je veľmi dobrou možnosťou. V tomto prípade sa snažíme vytvoriť regulárny výraz, ktorý nájde presne to, čo hľadáme. Pravda, platí to len vtedy, ak to, čo hľadáme, je tak jednoduché, že sa to dá opísať ako regulárny výraz. Problémom je však nevyspytateľnosť HTML kódu, v ktorom treba počítať s viacerými zlozvykmi jeho tvorcov.

Podľa definície by mal byť HTML kód veľmi podobný s klasickým XML. Opak je však pravdou, pretože, ako sme už povedali, HTML toleruje veľa zlozvykov a nedokonalostí. Práve preto je lepšie využiť knižnicu pripravenú práve pre prácu s HTML.

Ťažké je vybrať v tomto prípade najlepšiu možnosť. Ak by sme chceli využiť regulárne výrazy, bolo by napísanie takéhoto výrazu tak, aby zachytil všetky prípady, veľmi komplikované.

Ale v prípade, žeby sa nám to podarilo, bolo by takéto hľadanie rýchlejšie ako použitie univerzálnej knižnice, ktorá je stavané na väčšie množstvo problémov.

Práve z týchto dôvodov sme sa v ďalšom texte rozhodli využiť knižnicu BeautifulSoup [5]. Predpripravili ju na prácu s HTML a na náš účel nám veľmi dobre poslúži. S pomocou tejto knižnice sme si pripravili túto metódu, ktorá získava z kódu všetky adresy:

```
from bs4 import BeautifulSoup
from urllib.parse import urljoin
from collections import deque
import urlnorm

def get_links_from_html(html)
   soup = BeautifulSoup(html)
   links = soup('a')
   for link in links:
      link = urljoin(domain, link.get('href'))
      yield urlnorm.norm(link)
```

BeautifulSoup nie je súčasťou štandardnej inštalácie jazyka python a preto ju treba doinštalovať. Najskôr teda vytvoríme jej inštanciu, pričom do nej vložíme HTML kód danej stránky. Potom získame všetky adresy pomocou volania *soup(„a“)*, ktoré vyhľadá značku *a* vo všetkých častiach HTML kódu. Potom vkladáme adresu do kolekcie a vraciame jej výstup.

Yield

Kľúčové slovo *yield* tu slúži na postupné vracanie prvkov poľa. To znamená, že v prípade, ak sa funkcia zavolá a jej výstup sa priradí premennej, samotná funkcia sa ešte nespustí. Až keď začneme prechádzať prvky danej premennej, tak sa postupne začne vykonávať funkcia. Keď si vyžiadame prvý prvok, funkcia sa vykoná až po kľúčové slov *yield*. Následne vráti daný prvok a jej vykonávanie sa preruší. Keď si zase vyžiadame ďalší prvok z kolekcie, funkcia sa znova spustí po kľúčové slovo *yield* a vráti ďalší prvok, až pokým nie je vyčerpaná kolekcia. Takýmto spôsobom šetríme zdroje našej aplikácie.

8.3.3 Relatívne vs. absolútne adresy

Ako sme už povedali, internet môže byť nevyspytateľný a preto sa nedá očakávať, že sa všade budú nachádzať odkazy v rovnakej podobe. Hlavný problém je medzi relatívnymi a absolútnymi adresami. Relatívna adresa má podobu */stuff/index.html,* zatiaľ čo v absolútnej adrese je uvedená celá cesta k súboru, teda napr. *http://www.fi.it/stuff/index.html*. Relatívna adresa sa viaže na aktuálny dokument, na ktorom sa nachádzame a hovorí, akou cestou sa z tohto dokumentu dostaneme k inému dokumentu.

My však potrebujeme zo stránok vždy extrahovať aktuálne adresy. Z toho dôvodu sme v predchádzajúcom príklade použili knižnicu *urllib.parse* a jej funkciu *urljoin* [10]. Táto funkcia ako vstup získava adresu aktuálneho dokumentu, na ktorom sa nachádzame a adresu iného dokumentu, na ktorý sa chceme z tohto dokumentu dostať.

Výstupom z tejto funkcie je správna absolútna adresa k zadanému dokumentu. Teda ak sa budeme napríklad nachádzať na stránke http://fi.it/ a na nej objavíme adresu */stuff/index.html,*

funkcia nám z týchto parametrov vytvorí adresu *http://fi.it/stuff/index.html*. Hlavná výhoda je však v tom, že ak budú parametre *http://fi.it/* a *http://stuba.sk/stuff2/index.html,* funkcia nám vráti neporušenú adresu *http://stuba.sk/stuff2/index.html*. Vďaka tejto funkcii teda dostaneme vždy správnu adresu pre ďalšie spracovanie.

8.3.4 Normalizácia adries

Keď už máme adresy, ktoré sú v rovnakej podobe, ostáva nám posledný problém - identifikácia duplicitných adries. Problém dnešného webu je to, že existuje viacero podôb tej istej adresy. Napríklad adresa *www.fi.it* je to isté, ako *www.fi.it/index.html*, *http://www.fi.it* alebo *fi.it*. Okrem toho môžu mať adresy rozdiely vo veľkých a malých písmenách, alebo je na ceste adresy niekde použitý reťazec „..“ pre návrat do predchádzajúceho adresára, alebo mnoho rôznych ďalších odlišností.

My však potrebujeme to, aby sme adresu navštívili iba raz a nepridávali ju viackrát do zoznamu. To zabezpečíme procesom, ktorý sa nazýva normalizovanie. Normalizovanie zahŕňa:

- prevod na malé písmená,
- konvertovanie adresy do tvaru IDN (medzinárodné mená domén, angl. Internationalized domain name),
- vymazávanie predvoleného portu 80,
- zmenšovanie cesty (nahrádzanie znakov ./ ../ // a podobne),
- vymazávanie posledného znaku „.“ z cesty,
- výmena všetkých znakov % číslo znaku za konkrétny znak tam, kde je to možné,
- pri ostatných znakoch s % číslo znaku premieňa na veľké písmená,
- premena medzier na %20,
- normalizovanie IP adries.

Na to, aby sme pridali správne adresy, využívame knižnicu *urlnorm* [11]. V hore uvedenom kóde spúšťame normalizovanie príkazom: *urlnorm.norm(link)*. Na tento účel existuje veľké množstvo knižníc nielen pre python, ale aj pre iné programovacie jazyky. Nie je takisto problém spraviť si vlastné pravidlá pre normalizovanie, hoci v takom prípade je veľké riziko zabudnutia na niektorú z podmienok a teda vytvorenie nedokonalej normalizácie.

8.3.5 Zoznamy navštívených adries a adries na navštívenie

Ďalším problémom, s ktorým sa pri tvorbe preliezača stretneme, je ukladanie webových adries. Na tento účel potrebujem dve úložiská:

- Pre adresy, ktoré chceme navštíviť
- pre navštívené adresy.

Prvé úložisko potrebuje z dôvodu, že nie okamžite po nájdení ďalšej adresy ju ideme navštíviť. Adresy si musíme ukladať a podľa zvolenej stratégie ich ďalej navštevovať. Medzi najpoužívanejšie stratégie patria: stratégia do hĺbky, do šírky, stratégie založené na čiastočnom skóre stránok (angl. PageRank) a ďalšie.

Druhé úložisko je nutné z dôvodu kontroly navštívených stránok. Keby sme novonájdenú stránku nekontrolovali voči už navštíveným stránkam, ľahko by sa nám mohlo stať, že by sa nám náš preliezač dostal do nekonečného cyklu.

Jedným zo spôsobov, ako zabezpečiť tieto dva zoznamy, je využiť vstavané funkcie jazyka python a teda napríklad údajovú štruktúru zoznam. Pomocou nej implementovať adresy, ktoré sa ešte musia navštíviť a všetky navštívené adresy. V tomto prípade sa však veľmi rýchlo môžeme stretnúť s pamäťovým problémom a tiež rýchlosť nášho riešenia nebude optimálna.

Implementácia pomocou Berkeley DB

Druhou možnosťou je využitie niektorej z pokročilých údajových štruktúr. V ďalšom texte budeme preto využívať nástroj Berkeley DB [7] s rozšírením pre python. Berkeley DB [12] sa zaraďuje medzi vnorené databázy, pričom je pripravená na ukladanie párov kľúč-hodnota.

Na náš účel budeme potrebovať dve databázy, pričom každá z týchto databáz má mať odlišné vlastnosti. Pre potreby ukladania adries budeme potrebovať databázu typu FIFO, pretože ideme realizovať vyhľadávanie do šírky. Na rozdiel od toho v prípade zoznamu navštívených stránok nebudeme stránky vyberať, potrebujeme iba rýchly spôsob vkladania stránok a veľmi rýchle vyhľadávanie v uložených stránkach.

Naša implementácia pre zoznam stránok pre navštívenie vyzerá takto:

```
import bsddb

class QueueDB():
    def __init__(self, dbfile):
        self.database_file = db_file
        self.database = bsddb.db.DB(None,0)
        self.database.set_re_len( 512 )
        self.database.open( self.database_file,
                            dbname = None,
                            dbtype = bsddb.db.DB_QUEUE,
                            flags  = bsddb.db.DB_CREATE,
                            mode   = 0,
                            txn    = None, )

    def pop_url(self):
        url = self.database.consume()
        if url == None:
            return url
        url = url[1].strip()
        return url

    def push_url(self, url):
            self.database.append(url)

    def push_urls(self, url_list):
        for url in url_list:
            self.database.append(url)
```

Pri vytváraní triedy QueueDB treba vytvoriť príslušnú databázu pomocou rozšírenia Berkeley DB. V našom prípade ide o databázu s veľkosťou záznamu 512 znakov, pričom jej typ bude DB_QUEUE čiže front. Okrem toho v triede implementujeme tri funkcie. Jednu na získanie ďal-

šej adresy na navštívenie, pričom daná adresa sa vyraďuje zo zoznamu adries na navštívenie. Ďalšie funkcie slúžia na vloženie jednej a viacerých adries do našej databázy.

Naša implementácia zoznamu všetkých už navštívených adries bude vyzerať takto:

```
class DuplCheckDB():
    def __init__(self, dbfile):
        self.database_file = db_file
        self.database = bsddb.db.DB(None,0)
        self.database.open(self.database_file,
                           dbname=None,
                           dbtype=bsddb.db.DB_HASH,
                           flags=bsddb.db.DB_CREATE,
                           mode=0,
                           txn=None, )
    def is_url_dupl(self, url):
        return (self.database.get(str(url))==None)

    def add_url(self, url):
        self.database.insert(url, "")
        return True
```

V tejto implementácii vytvárame triedu DuplCheckDB, ktorá vytvára samotnú databázu. Táto trieda pri inicializácii vytvorí databázu typu DB_HASH. Vďaka tomu bude mať vyhľadávanie v tejto databáze zložitosť O(1). Horšie to bude v prípade vkladania údajov, kde bude zložitosť exponenciálna. Trieda má okrem toho ďalšie dve funkcie; prvú na zistenie, či sa daná adresa nachádza v databáze a druhú na vloženie adresy do databázy.

8.4 Spojenie všetkých častí

V predchádzajúcich kapitolách sme si pripravili triedy a metódy pre to, aby sme teraz mohli jednoducho vytvoriť výsledný proces preliezania webu. V nasledujúcom príklade teda ukazujeme celý obsah triedy Crawler [1], pričom definície metód, ktoré sme ukázali už v predchádzajúcich podkapitolách, vynechávame.

```
from database_wrappers include QueueDB
from database_wrappers include DuplCheckDB

def Crawler(object):

    def __init__(self):
        self.queue = QueueDB('queue.db')
        self.duplcheck = DuplCheckDB('duplcheck.db')


    def init_database(urls):
        self.queue.push_urls(urls)


    def get_page_content(pageName):
        ...


    def get_links_from_html(html):
        ...
    def crawl():
        while (1):
```

```
url = self.queue.pop_url()
try:
    if (url == none):
        print('preliezac spracoval všetky stránky')
        break
    pageContent = get_page_content(url)
    if (pageContent != none) :
        links = get_links_from_html(pageContent)
        for url in links:
            if(self.duplcheck.is_url_dupl(url):
                self.duplcheck.add_url(url)
                self.queue.push_url(url)
except (Exception, e):
    log(e)
```

V hore uvedenom príklade vidíme, že pri vytváraní objektu Crawler (metóda __init__) sa vytvárajú aj dve databázy QueueDB pre zoznam adries na navštívenie a DuplCheckDB pre zoznam navštívených adries. Zoznam adries na navštívenie je nutné naplniť pomocou metódy init_database.

Následne sa dostávame k metóde crawl, ktorá v nekonečnom cykle prechádza všetky adresy v zozname adries na navštívenie. Pokiaľ sa v našom zozname už nenachádzajú žiadne adresy, proces preliezania webu končí.

V opačnom prípade sa podľa zadanej adresy získava obsah stránky. Ak sa nám obsah podarí získať, vyťahujeme z neho všetky adresy. Adresy sa kontrolujú na duplicitnosť a následne sa vkladajú do zoznamu adries na navštívenie. Takto sa pokračuje až do chvíle, keď sa prelezú všetky adresy. Metóda pre spustenie takéhoto preliezača by vyzerala takto:

```
Import crawler

crawler = Crawler()
crawler.init_database(('www.fi.it', 'www.stuba.sk'))
crawler.crawl()
```

8.4.1 Kde hľadať vylepšenia

Náš preliezač má základnú funkcionalitu. K tejto funkcionalite sa toho však dá ešte veľa pridať. Bolo by napríklad veľmi dobre zmeniť spôsob vyberania novej adresy z radu tak, aby sa preliezané weby striedali a teda aby preliezač nevyťažoval jednu stránku niekoľkými požiadavkami za sekundu. Takisto by sa dala uplatniť niektorá metrika, napríklad čiastočný Page-Rank alebo HITS a podobne.

Ďalším rozšírením by mohlo byť paralelizovanie procesu preliezania stránok. Pri paralelizácii by program navštívil viacej stránok. Bolo by však nutné takisto ošetriť prístupy k zdieľaným databázam.

V úvodnej fáze sme ukázali, ako odfiltrovať všetok obsah, ktorý nie je HTML stránkou. No preliezače často potrebujú indexovať aj iný obsah, ako sú obrázky, pdf dokumenty alebo dokonca videá. Tu by sa dali uplatniť viaceré stratégie spracovania na základe typu sťahovaného dokumentu.

V neposlednom rade sme sa nezaoberali súborom *sitemap.xml*. Tento súbor slúži ako pomôcka od vlastníka domény pre preliezače a bol zavedený spoločnosťou Google v roku 2005.

Hovorí, ktoré časti stránky nie je nutné preliezať, ďalej hovorí o periodicite navštevovania stránok, ale aj o lokalitách, na ktoré chce preliezač upozorniť.

8.5 Zhrnutie

V tejto kapitole sme na začiatku opísali relatívne mladý jazyk python. Tu sme ukázali jeho veľmi zaujímavú vlastnosť použitia odsadení na označenie blokov. Následne sme opísali, čo je to webový preliezač a akým spôsobom s na dnešnom webe používa. Hlavná časť našej kapitoly potom patrila vytvoreniu webového preliezača v jazyku python. Tu sme poukázali na niektoré nástrahy, ktoré tu čakajú na vývojára. Ukázali sme spôsoby riešenia relatívnej a absolútnej cesty, spôsoby normalizácie adries, ale aj efektívne spôsoby vytvárania databáz pre zachytávanie zoznamov adries. V záverečnej časti sme skonštruovali základnú štruktúru preliezača.

Literatúra

[1] Easter eggs in Python. (2010). Dostupné na: http://digitizor.com/2010/05/02/easter-eggs-in-python/

[2] Guido van Rossum: The History of Python. (2013). Dostupné na: http://python-history.blogspot.sk/

[3] Python crawler. (2013). Dostupné na:

[4] Reitz K., Picking an Interpreter (2013). Dostupné na: http://docs.python-guide.org/en/latest/starting/which-python/

[5] Richardson L., Beautiful Soup. (2013). Dostupné na: http://www.crummy.com/software/BeautifulSoup/

[6] Lutz M.: *Programming Python*. O'Reilly Media, (2006).

[7] Oracle Berkeley DB. (2013). Dostupné na: http://www.oracle.com/technetwork/database/database-technologies/berkeleydb/overview/index.html

[8] Python Programming Language – Official Website. (2014). Dostupné na: www.python.org

[9] Urllib – Open arbitrary resurces by URL. (2013). Dostupné na: http://docs.python.org/2/library/urllib.html

[10] Urlparse – Parse URL into components. (2013). Dostupné na: http://docs.python.org/2/library/urlparse.html

[11] Urlnorm . (2013). Dostupné na: https://pypi.python.org/pypi/urlnorm

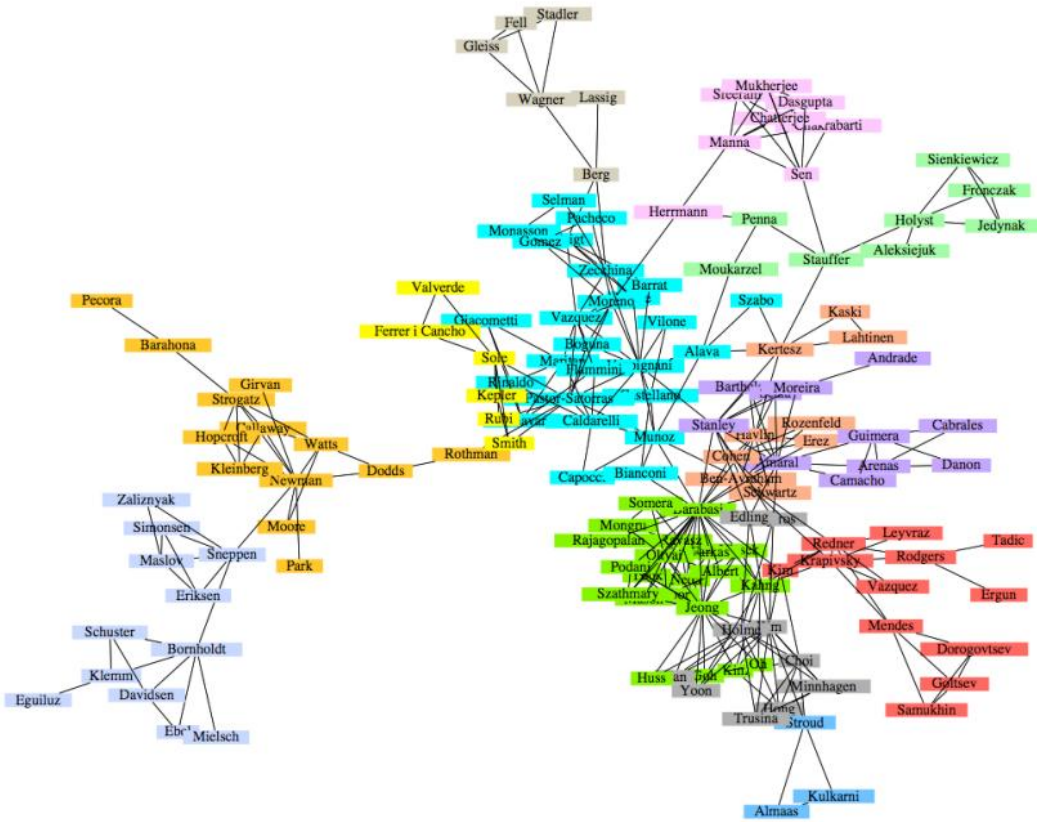
[12] Berkeley DB 3.x & 4.x Python Extension Package. (2013). Dostupné na: http://pybsddb.sourceforge.net/bsddb3.html

[13] Web Crawler. (2014). Dostupné na: http://en.wikipedia.org/wiki/Web_crawler

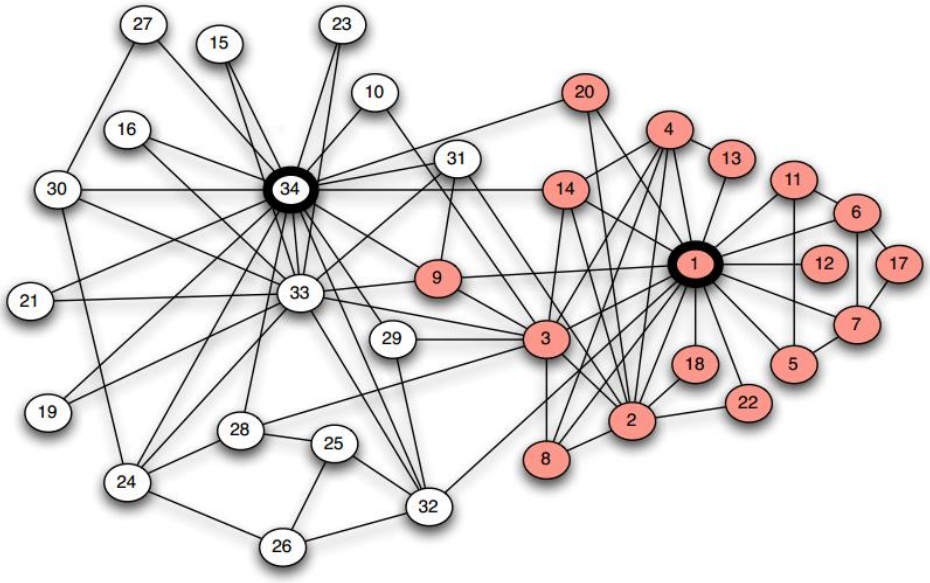
9 Rozdeľovanie grafov

V súčasnosti sa takmer vo všetkých oblastiach záujmov ľudstva vytvára veľké množstvo údajov. Tvorba týchto údajov je dôsledkom rozširovania a uľahčovania prístupu k internetu a tým aj prístupu k informáciám. Avšak dnes už sa o internete nehovorí len v oblasti informačných a komunikačných technológií ale aj o takzvanom internete vecí (angl. Internet of Things), kde internet preniká do najrôznejších oblastí nášho každodenného života. Pri takomto množstve údajov má zmysel snažiť sa vytvárať a hľadať nové spôsoby, metódy a algoritmy na prácu s nim ako napríklad ich rozdeľovanie alebo zoskupovanie. Rozdeľovanie grafov zahŕňa práve niekoľko takýchto metód.

S pojmom veľké údajové korpusy (angl. Big Data) sa v dnešnej dobe stretávame už pomerne často. Je to populárny pojem opisujúci exponenciálny nárast údajov a ich dostupnosť (štruktúrované aj neštruktúrované údaje) [3]. So súčasnými možnosťami jednoduchej komunikácie a spolupráce v rámci celého sveta sa nové údaje generujú obrovskou rýchlosťou a v obrovskom množstve. Významným zdrojom nových údajov sú webové systémy poskytujúce služby pre sociálne siete, komunitné weby, weby otázok a odpovedí (angl. Community Question Answering) alebo aj weby zamerané na vedu a výskum a ďalšie. V takýchto sieťach často potrebujeme identifikovať pevne zoskupené skupiny (Obrázok 30) alebo zisťovať ako ich rozdeliť (Obrázok 31). Siete a vzájomné vzťahy ich položiek sa dajú reprezentovať pomocou grafov, kde uzly grafu predstavujú položky siete a hrany medzi nimi ich vzájomný vzťah. V sociálnych sieťach uzly môžu reprezentovať napríklad osoby a hrany ich vzájomné priateľstvo. V sieti spoluautorov uzly predstavujú osoby a hrany spoločnú publikáciu. Na identifikovanie skupín v takýchto grafoch alebo ich rozdelenie je vhodné použiť niektoré metódy rozdeľovania grafov [2].



Obrázok 30. Sieť spoluautorov - Ako môžu byť pevne zoskupené skupiny identifikované [1] ?



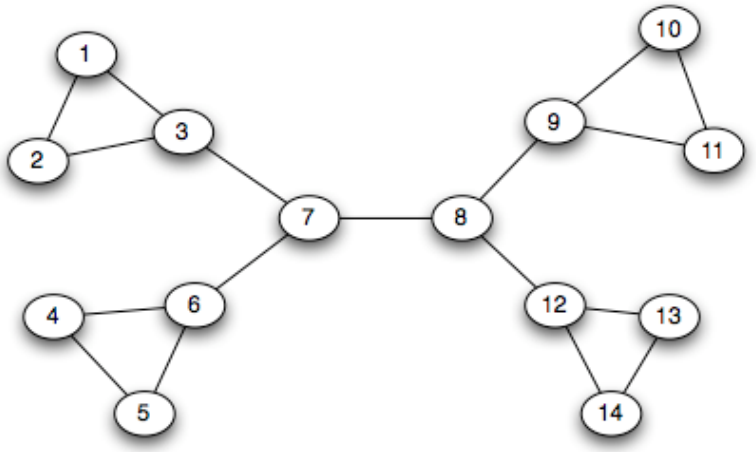
Obrázok 31. Karate klub sa rozdelí po hádke prezidenta (34) a inštruktora (1) - Môžu byť nové kluby identifikované na základe siete [1]?

9.1 Metódy rozdeľovania grafov

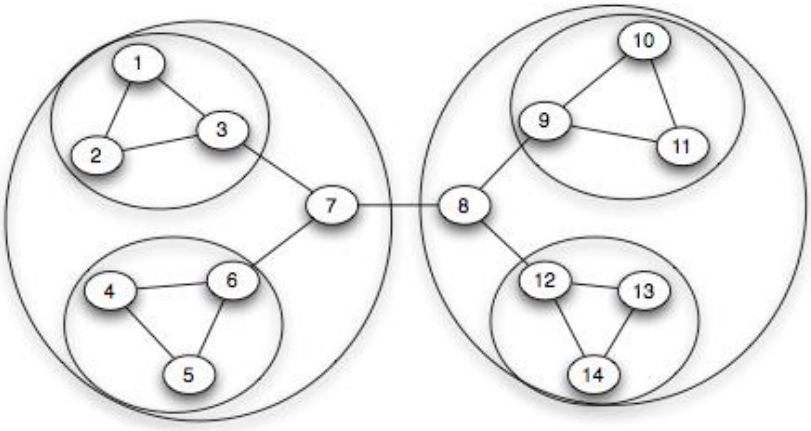
Pre problém rozdeľovania grafov a sietí do pevne zoskupených regiónov navrhli mnoho rôznych prístupov [2]. Často existuje široké množstvo efektívnych metód použiteľných na konkrétny problém.

Jedna skupina z týchto metód sa sústreďuje na identifikáciu a odstraňovanie spojov (hrán) medzi husto prepojenými oblasťami. Po odstránení týchto spojov sa sieť začne rozpadať na podsiete, v ktorých môžu byť znovu identifikované ďalšie spoje na odstránenie a proces pokračuje. Tieto metódy sa nazývajú deliace metódy (angl. Divisive).

Alternatívna skupina metód začína z opačného konca problému a zameriava sa na pevne zoskupené časti siete. Táto metóda hľadá uzly, ktoré pravdepodobne patria do rovnakého regiónu a spája ich. Po skončení tohto procesu existuje veľké množstvo pospájaných častí obsahujúcich husto prepojené regióny. Tejto metóde sa hovorí aglomeratívna metóda (angl. Agglomerative).



Obrázok 32. Ukážková sieť [2].



Obrázok 33. Pevne zoskupené skupiny a ich vnorená štruktúra [2].

Graf na obrázku 32 možno intuitívne rozdeliť do regiónov zobrazených na obrázku 33. Ako vidno, tak je možné graf rozdeliť do oblasti (regiónu) skladajúcej sa z uzlov 1-7 a oblasti skladajúcej sa z uzlov 8-14. V rámci týchto regiónov je možné ďalšie rozdelenie: na ľavej strane s uzlami 1-3 a 4-6, na pravej strane 9-11 a 12-14.

9.2 Medzipoloha (angl. Betweenness)

Definícia: *Medzipoloha hrany* je celkové množstvo najkratších ciest zo všetkých uzlov do všetkých ostatných uzov prechádzajúcich cez danú hranu.

Definícia: *Medzipoloha uzla* je celkové množstvo najkratších ciest zo všetkých uzlov do všetkých ostatných uzlov prechádzajúcich cez daný uzol.

Príklad: Uvažujme hranu 7-8 z obrázku 32. Pre každý uzol A na ľavej polovici grafu a pre každý uzol B na pravej polovici grafu prechádza tok cez hranu 7-8. Medzi pármi uzlov, ktoré ležia na rovnakej polovici neprechádza žiadny tok medzi ich hranami. Keďže na každej polovici grafu je 8 hrán, tak výsledok medzipolohy pre hranu 7-8 je 8 * 8 = 64. Rovnakým spôsobom sa určia hodnoty medzipolohy pre všetky zvyšné hrany.

9.3 Girvanov-Newmanov algoritmus

Algoritmus, ktorý navrhli Girvan a Newman v roku 2002 je deliaca metóda, ktorá sa v posledných rokoch široko využíva najmä pre údaje zo sociálnych sietí [1, 2]. Táto metóda úspešne odstraňuje hrany s vysokou medzipolohou a môže byť zhrnutá nasledujúco:

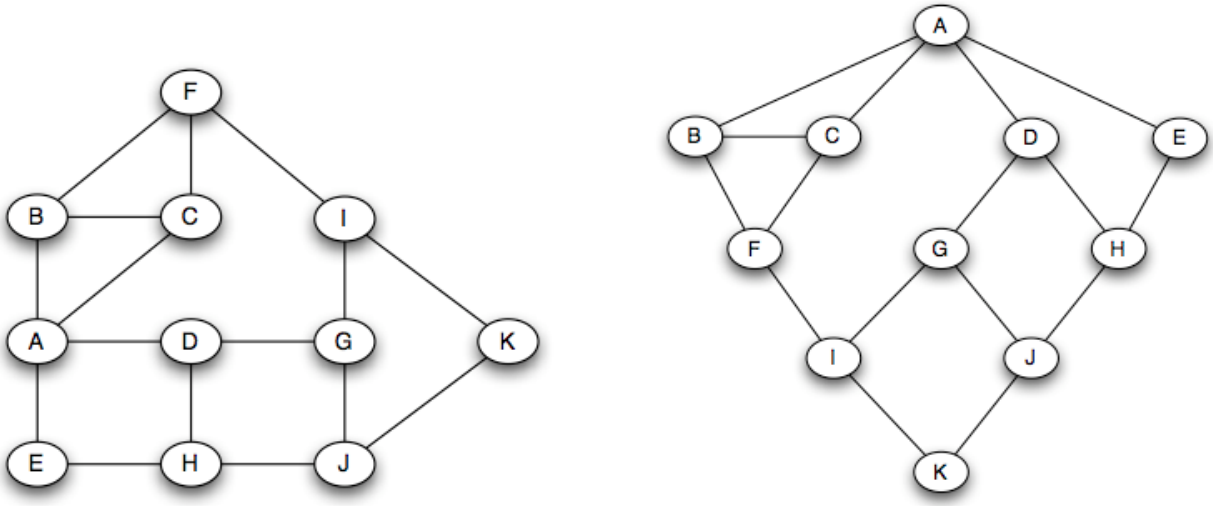
1. Nájsť hranu/y s najvyššou medzipolohou a odstrániť túto/tieto hranu/y z grafu. Toto môže spôsobiť rozdelenie grafu na niekoľko komponentov, čím vzniknú nové regióny.
2. Prepočítať všetky medzipolohy, a znovu odstrániť tie s najvyššou medzipolohou. Veľké komponenty sa môžu rozdeliť na menšie čím vzniknú nové vnorené regióny.
3. Týmto spôsobom pokračovať ďalej kým nezostanú žiadne hrany alebo nedosiahneme požadovaný počet regiónov.

9.4 Výpočet hodnôt medzipolôh

Zložitou časťou Girvanovej-Newmanovej metódy je, že definícia medzipolohy zahŕňa úvahy o množine všetkých najkratších ciest medzi dvojicami uzlov v [1, 2]. Keďže môže existovať veľké množstvo takýchto ciest, tak výpočet medzipolôh je náročný. Existuje však šikovný spôsob, ako počítať medzipolohy, efektívne založený na vyhľadávaní do šírky. V jednom čase sa uvažuje graf z perspektívy jedného uzlu. Pre všetky uzly sa vypočíta, ako je celkový tok z aktuálneho uzlu distribuovaný cez hrany. Ak sa to urobí pre každý uzol, tak je možné jednoducho spočítať všetky toky a tak získať výslednú medzipolohu na každej hrane.

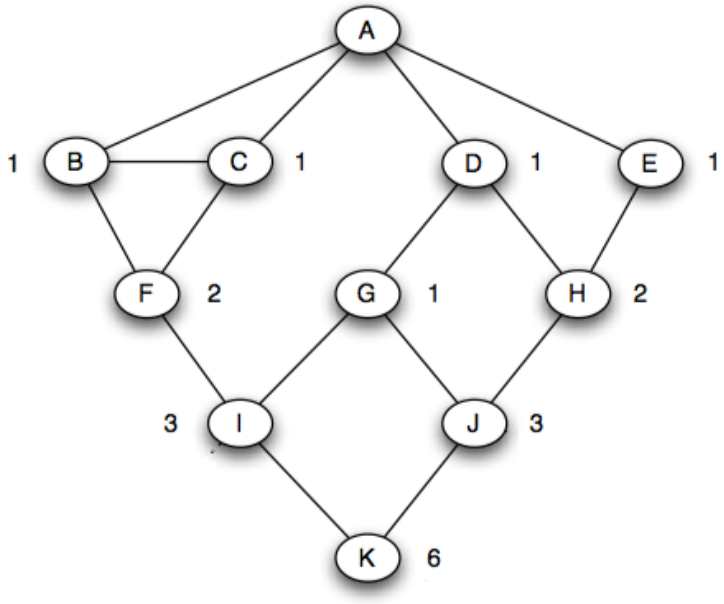
Príklad: Ukážková sieť Obrázok 34

1. Vykonať vyhľadávanie do šírky so začiatkom vo uzle A (Obrázok 34 - vpravo)
2. Určiť počet najkratších ciest z A do každého iného uzlu (Obrázok 35)
3. Na základe počtu najkratších ciest určiť množstvo toku z A do všetkých ostatných uzlov ktoré používajú každú hranu (Obrázok 36)



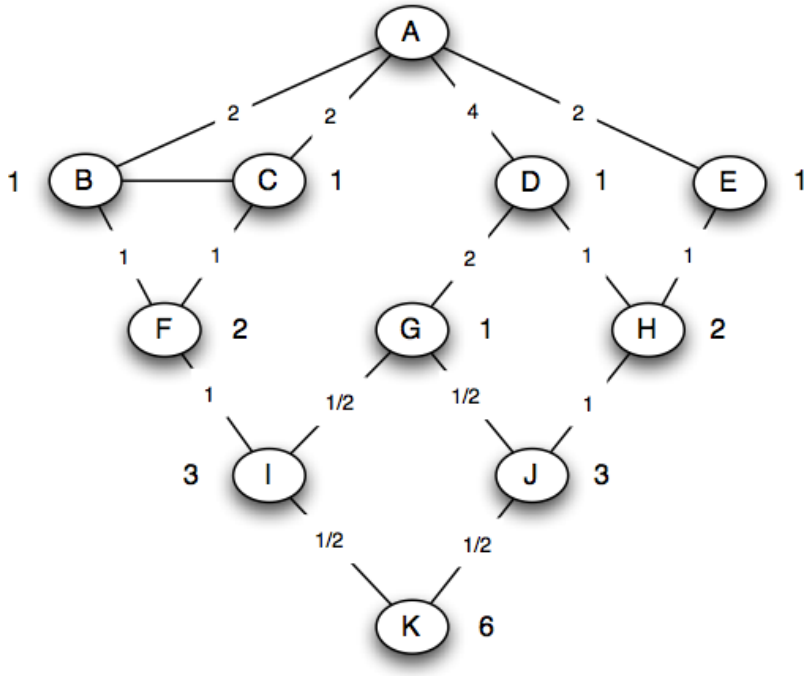
Obrázok 34. Ukážková sieť (vľavo), sieť prehľadávania do šírky z uzlu A (vpravo) [2].

Spočítanie najkratších ciest idúcich do uzlu sa dá urobiť pomerne jednoducho spočítavaním najkratších ciest z predchádzajúcich uzlov pri pohybe smerom do nižších vrstiev v grafovej štruktúre pre prehľadávanie do šírky (Obrázok 35).



Obrázok 35. Početnosť najkratších ciest z uzlu A [2].

Posledný krok (určenie množstva toku) sa začína z najnižšej vrstvy grafovej štruktúry pre prehľadávanie do šírky a pokračuje smerom nahor. Tu sa tok rozdeľuje k uzlom vo vyššej vrstve proporcionálne podľa počtu najkratších ciest vedúcich do uzlu (Obrázok 36).



Obrázok 36. Určenie množstva toku pre jednotlivé hrany z uzlu A [2].

9.5 Zhrnutie

Pojmu veľké údajové korpusy (Big Data) sa venuje čím ďalej väčšia pozornosť a vývoj v oblasti informačných a komunikačných technológií nasvedčuje, že to bude pokračovať. Metódy venujúce sa rozdeľovaniu grafov pomáhajú práve pri práci a analýze veľkého objemu údajov tým, že ich rozdeľujú (resp. zoskupujú) do menších celkov na základe nejakých spoločných vlastností. V tejto kapitole sme si povedali o deliacich a aglomeratívnych metódach rozdeľovania grafov. Konkrétne sme sa venovali tzv. medzipolohe a objasnili sme si Girvanov-Newmanov algoritmus. S vďakou uvádzame, že text tejto kapitoly vychádza okrem iných z obsahu prednášky, ktorej autorom je McCrown [1].

Literatúra

[1] McCown, F.: Introduction to Web Science. Harding University, (2013). Dostupné na: http://www.harding.edu/fmccown/classes/comp475-s13

[2] Easley, D., Kleinberg J.: *Networks, Crowds and Market: Reasoning About a Highly Connected World*. Cambridge, (2010). ISBN: 9780521195331. Dostupné na: http://www.cs.cornell.edu/home/kleinber/networks-book/

[3] Big Data, What it is and why it matters. (2013). Dostupné na: http://www.sas.com/en_us/insights/big-data/what-is-big-data.html

10 Sociálne siete a v nich prítomné mechanizmy

Sociálna sieť sa stala často skloňovaným pojmom vďaka populárnym webovým aplikáciám poskytujúcim služby sociálnych sietí. Veľké množstvo údajov o prepojeniach medzi jednotlivcami získané z týchto aplikácií umožňuje rozsiahle analýzy sociálnych sietí, ktoré nám poskytujú overenie rôznych teoretických modelov. Kontext, v ktorom sa nachádza sociálna sieť, má veľký vplyv na jej štruktúru. Každý jedinec má nejaké charakteristiky a podobnosť týchto charakteristík medzi dvoma ľuďmi ovplyvňuje, či sa medzi nimi vytvorí vzťah. Veľmi často sa vytvárajú prepojenia práve medzi ľuďmi, ktorí sú si podobní svojimi záujmami a zúčastňujú sa rovnakých činností. Podobne, každý príslušník sociálnej siete je formovaný správaním a činnosťami svojich kamarátov. Vďaka poznatkom získaným štúdiom sociálnych sietí možno zefektívniť rôzne webové služby ako napríklad vyhľadávanie, odporúčanie atď.

Pojem sociálna sieť v posledných desaťročiach púta veľký záujem o výskum v spoločenských vedných disciplínach. Sociálna sieť na webe je štruktúra vytvorená z uzlov, reprezentujúcich napríklad jednotlivcov a organizácie, nad ktorými sú definované vzťahy. Pozornosť sa sústreďuje predovšetkým na analýzu vzťahov medzi entitami v sieti a na ich vzory. Očakáva sa, že ich analýza môže priniesť jasnejší pohľad na politické, ekonomické a sociálne prostredie [11].

V súčasnej informatizovanej spoločnosti sa denne stretávame s pojmom sociálna sieť, ktorou sa zvyčajne populárne aplikácie Facebook, či Twitter. Sociálna sieť je omnoho všeobecnejší pojem a v tomto prípade sa pojem sociálna sieť zamieňa za online webové sídla poskytujúce služby sociálnych sietí. Tieto sídla sociálnych sietí jednotlivcom umožňujú: vytvoriť verejný alebo čiastočne verejný profil v rámci systému, vytvárať prepojenia s inými používateľmi a pre-

chádzať zoznam ich prepojení alebo prepojení vytvorených ostatnými v systéme [13]. Tieto stránky sa však sústreďujú najmä na poskytovanie rôznych služieb, na čo využívajú sociálnu sieť ich používateľov.

Sídla sociálnych sietí sa v súčasnosti tešia obrovskej popularite. Službu Facebook si registrovalo 1.4 miliardy používateľov[28] [12]. Takéto veľké množstvo údajov o štruktúre prepojení, dostupných vďaka sídlam sociálnych sietí, umožňuje testovanie teórií o sociálnych vzťahoch oveľa presnejšie a vo veľkej miere. Zaujímavá je napríklad štúdia týkajúca sa mobilizovania voličov do volieb, ktorá mala 61 miliónov účastníkov, získaných prostredníctvom služby Facebook [17]. Použitím takýchto sídiel poskytujúcich služby sociálnych sietí sa môžu robiť rôzne sociálne štúdie s neporovnateľne väčšími rozmermi ako to bolo možné doteraz.

10.1 Štúdium sociálnych sietí

Štúdium sociálnych sietí sa pokúša vysvetliť, ako sú účastníci siete navzájom prepojení, ako to vplýva na ich správanie (z pohľadu na prepojenia) a ako ich interakcia ovplyvňuje štruktúru celej siete (z pohľadu na štruktúru). Dvoma základnými analytickými otázkami sú [10]:

- Prečo a ako účastníci siete navzájom interagujú pozorovaným spôsobom?
- Aké sú dôsledky pozorovanej štruktúry siete?

10.1.1 Praktické aplikácie

Štúdium sociálnych sietí sa s výhodou aplikuje v praxi. Prebieha mnoho výskumu s cieľom získavať a študovať poznatky o nich pre lepšie pochopenie správania sa alebo preferencií používateľov v rámci skupiny. Analýza ďalej umožňuje zlepšiť odporúčanie, filtrovanie, vyhľadávanie alebo zefektívnenie marketingu.

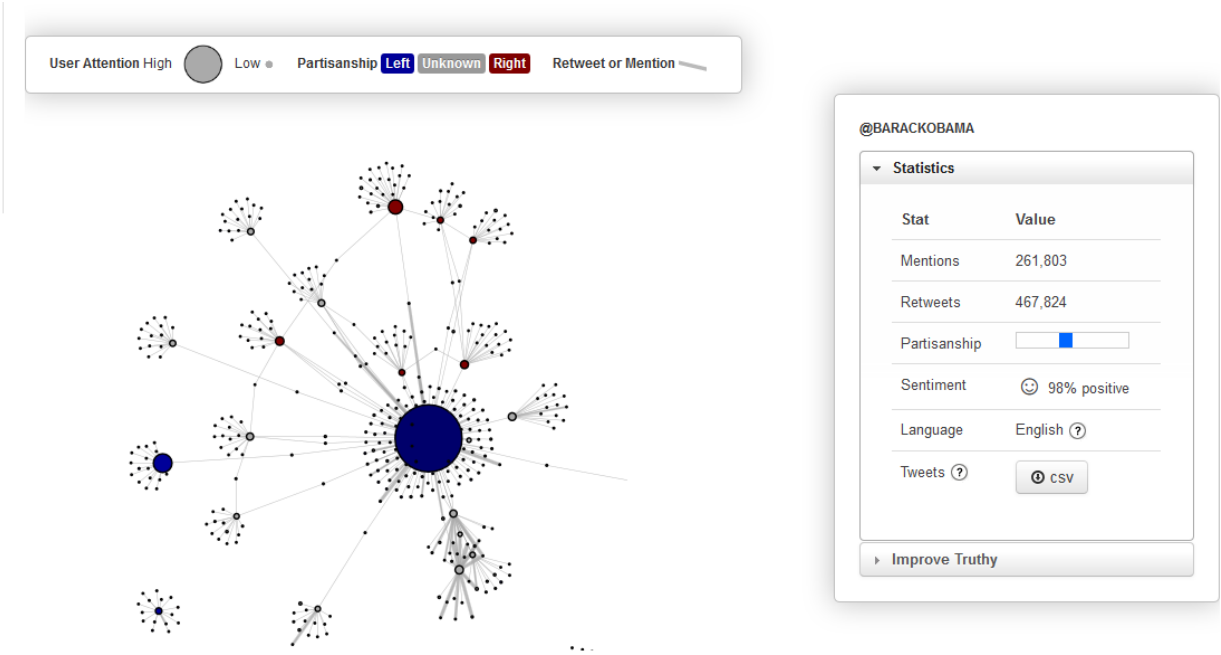
Google využíva svoju sociálnu sieť na vyhľadávanie tak, aby zvýšil relevanciu nájdených odkazov. Používatelia majú možnosť ohodnotiť stránky alebo reklamy kliknutím na tlačidlo „+1“, čím vyjadria, že daný odkaz je užitočný. Keď potom iný používateľ vyhľadáva, odkazy ohodnotené jeho kamarátmi zo sociálnej siete sú vo vyhľadávaní vyššie. Zároveň sa zobrazí, ktorý kamarát stránku ohodnotil, čo zvyšuje dôveryhodnosť týchto výsledkov vyhľadávania alebo reklám. Ak teda používateľ vyhľadáva informácie o zimnej lyžovačke a jeho priateľ, ktorý sa lyžovaniu venuje profesionálne, ohodnotil nejaký odkaz, zrejme bude pre neho vysoko relevantný [14].

Zaujímavý nástroj slúžiaci na analýzu sociálnych sietí je nástroj Truthy, ktorý dokáže analyzovať šírenie informácií, či už politických alebo obchodných, na stránkach sociálnej siete Twitter. Tento systém vyhodnocuje tisíce správ za hodinu kvôli rozpoznávaniu vzniku nových trendov. Systém obsahuje rámec pre problém predpovedania kľúčových ukazovateľov trhu a podporuje tak rozhodnutia pri obchodovaní [16].

Hlbšie poznatky o šírení informácií v sociálnych sieťach sa môžu použiť na podporenie šírenia konkrétnej informácie napríklad o produkte a teda podporiť jeho tzv. virálny marketing. Ak

[28] Informácia k 1.1. 2014.

sa zákazníkovi produkt nepáči, má negatívny potenciál pre marketing. Ak má zákazník veľký vplyv na svoje okolie (napríklad je tzv. celebrita), môže významne rozšíriť mienku o produkte a má tak vysoký potenciál pre zacielenie marketingu. Ak má zákazník síce málo prepojení, ale je medzi nimi vplyvná osoba - autorita, má taktiež veľký potenciál [15].



Obrázok 37. Vizualizácia šírenia správ od amerického prezidenta na sieti Twitter pomocou nástroja Truthy. Automaticky sa rozpoznáva príslušnosť k politickej ľavici/pravici [16].

10.1.2 Základné pojmy pre štúdium sietí

Analýza sociálnych sietí využíva pojmy ako účastník, väzba, dyáda, triáda, podskupina, skupina, vzťah a sociálna sieť. Spoločenské entity sa označujú ako *účastníci*. Sú to jednotlivé alebo kolektívne jednotky spoločnosti ako napríklad ľudia v skupinách, oddelenia vo firmách, či mestá v štátoch. Sociálna sieť sa skladá väčšinou z účastníkov jedného druhu.

Vlastnosťou *väzby* je, že vytvára prepojenie medzi dvoma účastníkmi. Väzba môže byť rôzneho druhu, ako napríklad: ohodnotenie osoby inou (priateľstvo, rešpekt), prevod materiálnych zdrojov (obchodné transakcie, pôžičky), asociácie a afiliácie (príslušnosť do záujmového klubu), interakcia (rozprávanie sa, posielanie správ), biologické a rodinné vzťahy (príbuzenstvo), fyzické prepojenie (cesta, most).

Na tej najzákladnejšej úrovni sa prepojenie vytvorené medzi dvoma účastníkmi nazýva *dyáda*. Je vlastnosťou dvojice, nielen jedného z účastníkov. Analyzuje sa reciprocita väzby alebo súčasný výskyt rôznych typov väzieb v sieti.

Triáda je množinou troch účastníkov a existujúcich alebo potenciálnych väzieb medzi nimi. Analyzuje sa najčastejšie tranzitívnosť vzťahov (ak účastník I má rád účastníka J a účastník J má rád K, tak aj I má rád K) alebo vyváženosť vzťahov (ak I sa má rád s J, tak sú podobní s účastníkom K).

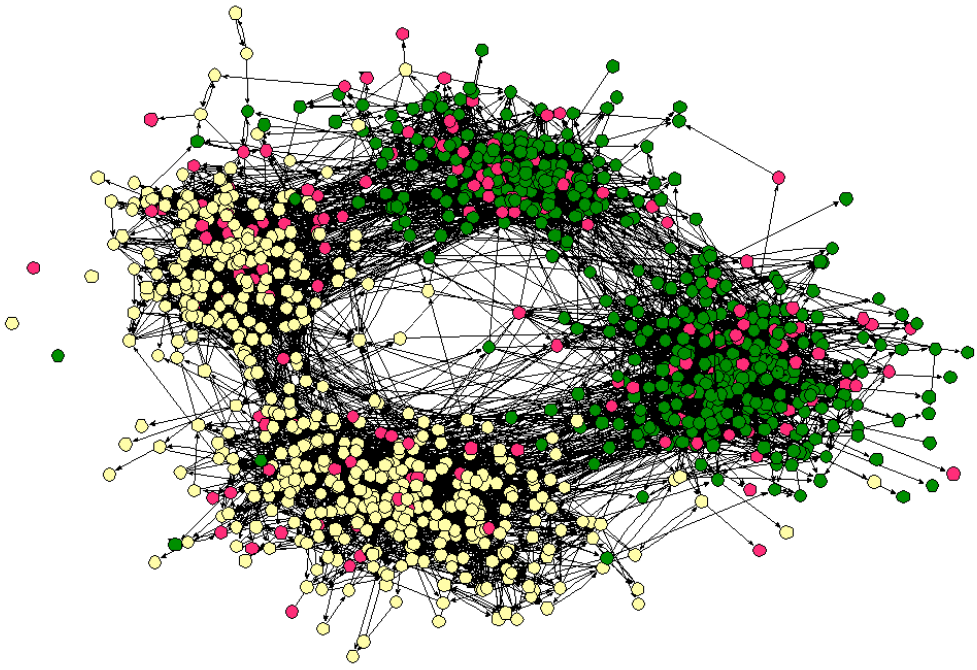
Analýza sietí sa zaoberá aj väčšími súbormi účastníkov, a teda vzťahmi medzi systémami účastníkov. Často sa vyberá konečná množina účastníkov, *skupina*, pre štúdium siete.

Súbor väzieb medzi účastníkmi skupiny sa nazýva *vzťah*. Príkladom je množina kamarátstiev žiakov v rámci triedy, kde väzby určujú vzťahy. Pre skupinu účastníkov je možné skúmať niekoľko vzťahov. V spomenutej školskej triede by to nemuseli byť len kamarátstva, ale napríklad aj to, kto komu čo požičal [11].

10.2 Homofília v sociálnych sieťach

Jedným zo základných javov v sociálnych sieťach je homofília, teda princíp, že máme tendenciu byť podobní so svojimi kamarátmi. Väčšina našich kamarátov sa nám podobá v nejakej charakteristike, napríklad vek, miesto kde žijeme, povolanie, záujmy a názory. Toto pozorovanie spomína už Platón vraviac, že „podobnosť plodí priateľstvo“ alebo Aristoteles, že „ľudia majú radi tých, ktorí sú ako oni sami“. Homofília poskytuje základný pohľad na to, ako sa formujú linky v sociálnych sieťach. Kontextuálne faktory, teda geografická blízkosť, rodina, organizácia, sociálna vrstva, to všetko vplýva na štruktúru siete a podnecuje vytváranie homofílnych prepojení. Homofília tak limituje sociálne svety ľudí na základe toho, aké informácie dostávajú, či aké postoje zaujímajú [2].

Príklad vplyvu kontextu na štruktúru siete je zachytený na obrázku 38 [7]. Táto sociálna sieť zobrazuje vzťahy medzi študentmi z dvoch základných a z dvoch stredných škôl ako aj ich rasovú príslušnosť. Dajú sa rozpoznať dve rozdelenia. Jedno sa zakladá na rase a rozdeľuje sieť na ľavú a pravú polovicu, druhé sa zakladá na príslušnosti ku strednej, resp. základnej škole a rozdeľuje na hornú a dolnú polovicu.



Obrázok 38. Sociálna sieť žiakov dvoch základných a dvoch stredných škôl. Farba uzlov vyjadruje rasovú príslušnosť žiakov [7].

10.2.1 Dôkaz prítomnosti homofílie v sieti

Pre danú charakteristiku sa dá testovať, či sa sieť vyznačuje homofíliou. Ak máme napríklad charakteristiku pohlavie, každý uzol v grafe predstavuje jedinca mužského pohlavia s pravdepodobnosťou p a ženského s pravdepodobnosťou q. Hrana medzi dvoma mužmi existuje s pravdepodobnosťou p^2, medzi dvoma ženami q^2 a medzi mužom a ženou alebo naopak s pravdepodobnosťou 2pq. Pri spomenutom teste spočítame množstvo hrán medzi jedincami s rôznym pohlavím a ak je tento počet významne menší ako hodnota 2pq, tak sa dá považovať existencia homofílie v sieti za preukázanú. V opačnom prípade, teda ak je významne väčší počet hrán medzi uzlami rovnakého pohlavia, ide o opačnú homofíliu [2].

10.2.2 Mechanizmy homofílie – selekcia a sociálny vplyv

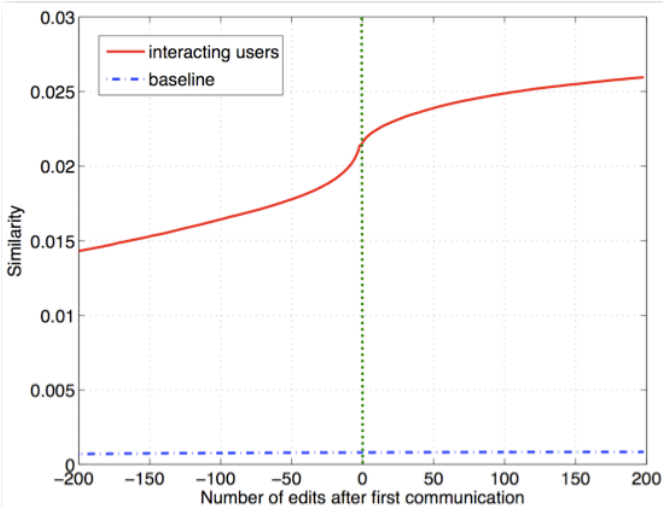
Vzťahy medzi ľuďmi s podobnými charakteristikami sú výsledkom dvoch hlavných mechanizmov, ktoré sa skrývajú za ich utváraním. Prvým mechanizmom je *selekcia*. Ľudia si vyberajú kamarátov podľa nemenných spoločných charakteristík. Sociálne prostredie dáva príležitosti vyutvárania vzťahov, napríklad medzi ľuďmi, ktorí bývajú v jednej štvrti, chodia do tej istej školy alebo práce. Premenlivé charakteristiky, ako napríklad správanie, činnosti a záujmy utvárajú zložitejšie spojenia v sieti. Okrem selekcie je tu prítomný druhý mechanizmus *socializácie* (*sociálneho vplyvu)*. Ľudia majú tendenciu upraviť svoje správanie tak, aby sa viac priblížilo správaniu ich kamarátov. Na sociálny vplyv sa tak možno pozerať ako na opak selekcie. To znamená, že pri selekcii sa utvárajú nové vzťahy na základe charakteristík, zatiaľ čo pri sociálnom vplyve existujúce spojenia utvárajú správanie [1, 8].

Základnou a skúmanou otázkou je, či ľudia v sieti prispôsobili svoje správanie, aby sa viac priblížili ich priateľom alebo či vyhľadali ľudí, ktorí už sú im podobní. Typický príklad pochádza z prostredia dospievajúcich ľudí a ich školských úspechov alebo drogových skúseností. Prítomné sú obidva mechanizmy. Tínedžeri hľadajú sociálne skupiny im podobné a tlak rovesníkov spôsobuje ich prispôsobenie správania sa skupine. Ktorý z týchto mechanizmov sa uplatňuje viac, je však náročnou a vo všeobecnosti nezodpovedanou otázkou.

Štúdia Christakisa [4] sa zaoberá vzájomným pôsobením mechanizmov medzi obéznymi ľuďmi. Autori skúmali tri dôvody pre zhlukovanie obéznych a zdravých ľudí osobitne: (i.) Je dôvodom selekcia, teda že ľudia si hľadajú kamarátov s podobným statusom obezity? (ii.) Sú dôvodom iné spoločné charakteristiky ľudí, ktoré súvisia so statusom obezity? (iii.) Je dôvodom zmena statusov obezity kamarátov s vplyvom na status jednotlivca? Prejavili sa všetky tri dôvody a ako najvážnejší sa však ukázal posledný, ktorý vyjadruje, že obezita sa môže šíriť pôsobením sociálneho vplyvu.

V prostredí webu sa otázkou vzájomného pôsobenia mechanizmov homofílie zaoberá výskumná práca [3]. V experimente sledovali správanie sa autorov článkov Wikipédie. V sociálnej sieti je medzi autormi spojenie, ak spolu navzájom komunikovali a ich správanie sa vyjadruje množstvo spoluvytváraných článkov. Podobnosť dvoch autorov sa definuje ako pomer množstva článkov spoluvytváraných oboma autormi a množstva článkov vytváraných aspoň jedným z autorov. Na obrázku 39 je graf znázorňujúci závislosť príbuznosti autorov od času, pričom čas

sa vyjadruje počtom editácií od prvej spoločnej komunikácie. Podobnosť dvoch autorov výrazne stúpa tesne pred prvou komunikáciou, keď sa prejavuje mechanizmus selekcie. Po prvej komunikácií podobnosť stále stúpa, nie však tak strmo, ako pred ňou. Tu sa prejavuje mechanizmus sociálneho vplyvu. Zdá sa, že v tomto prípade prevláda selekcia nad sociálnym vplyvom.



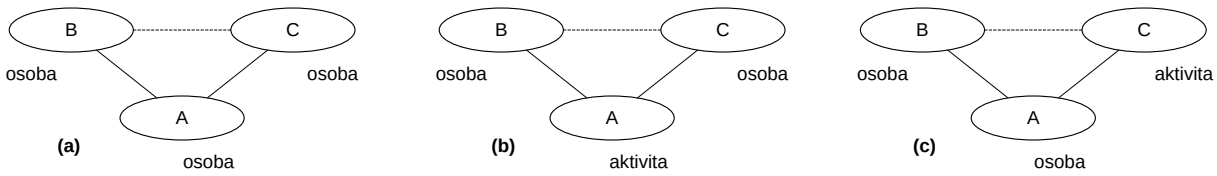
Obrázok 39. Príbuznosť autorov článkov v závislosti od času. Os x znázornňuje čas vyjadrený ako počet editácií od prvej spoločnej komunikácie [3].

10.3 Sociálno-afiliačná sieť

V súvislosti s utváraním prepojení v sociálnej sieti sa používa pojem triádový uzáver (z angl. triadic closure). Znamená to, že ak osoba B a osoba C má spoločného kamaráta, osobu A, tak sa zvyšuje pravdepodobnosť, že B a C sa stanú kamarátmi. Princíp homofílie hovorí, že ak dvojice A-B a A-C majú veľkú podobnosť, tak je pravdepodobné, že aj B a C sú si podobní. A takéto kamarátstvo sa môže vytvoriť aj keď si nie sú vedomí, že majú spoločného priateľa A [9].

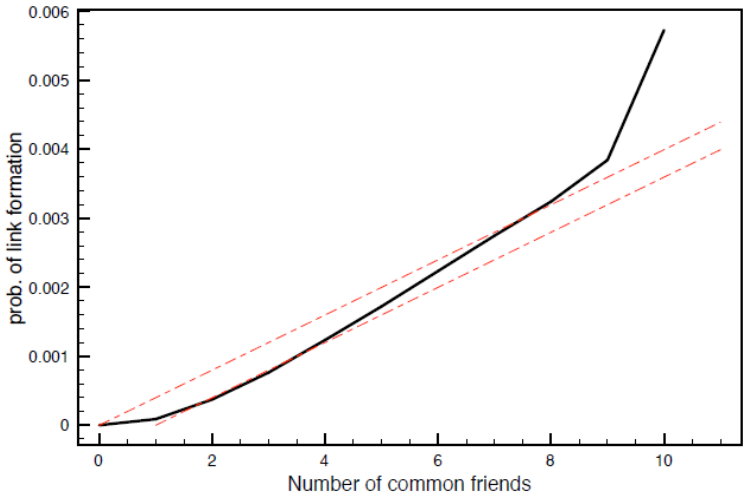
Okolitý kontext, v ktorom sa vytvárajú priateľstvá, možno taktiež reprezentovať v samotnej sieti, aby sme mali bližší pohľad na tento proces. Graf pozostáva teda z uzlov reprezentujúcich ľudí a uzlov reprezentujúcich činnosti, ktorých sa zúčastňujú. Hrany existujú medzi dvoma osobami a medzi osobou a činnosťou. Takáto sieť sa nazýva sociálno-afiliačná sieť.

V tomto type siete ide o niečo zložitejší proces uzáveru. Máme uzly B a C a ich spoločného suseda A. Predpokladáme, že nové spojenie sa vytvára medzi B a C. Môžu nastať tri rôzne situácie podľa toho, o aký typ uzla ide. (i.) Ak A, B aj C sú osoby, tak ide o klasický *triádový uzáver*. (ii.) Ak B a C reprezentujú ľudí a C je činnosť, tak ide o prípad selekcie, keď sa utvára spojenie medzi ľuďmi so spoločnou charakteristikou. Nazýva sa *záujmový uzáver* (z angl. focal closure). (iii.) Ak A a B sú ľudia a C je činnosť, ktorej sa zúčastňuje človek B a utvára sa spojenie medzi človekom a činnosťou, tak ide o prípad sociálneho vplyvu. Nazýva sa *členský uzáver* (z angl. membership closure). Všetky tri typy sú znázornené na obrázku 40 [1, 8].



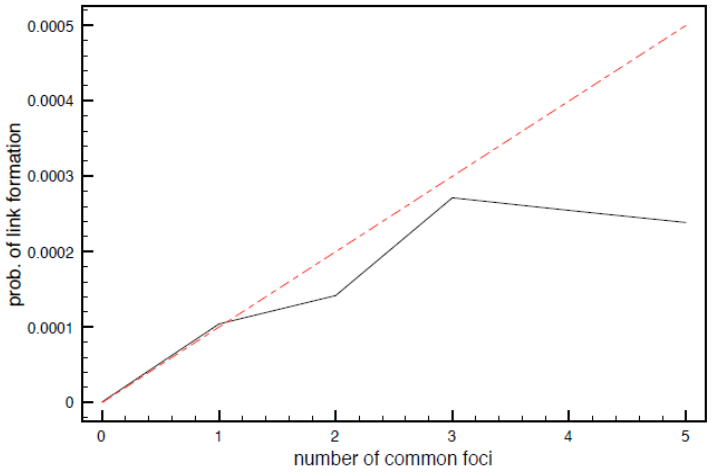
Obrázok 40. Triádový uzáver (a), ústredný uzáver (b), členský uzáver (c).

Kossinets [5] preskúmal, ako množstvo spoločných priateľov ovplyvňuje pravdepodobnosť vytvorenia prepojenia (triádový uzáver). Sledovali komunikáciu e-poštou medzi 22 000 študentmi počas doby jedného roka. Ak si študenti vymenili čo len jednu e-správu v priebehu 60 dní, tak sa považovali za prepojených. Autori porovnávali snímky siete z jednotlivých dní. V dvoch za sebe idúcich snímkach pozorovali zmenu v počte dvojíc uzlov, medzi ktorými sa nevytvorilo spojenie, a ktoré mali práve *k* spoločných priateľov. Keďže tieto snímky sa vytvárali každý deň, výsledok experimentu vyjadruje pravdepodobnosť vytvorenia spojenia za dobu jedného dňa. Na obrázku 41 je znázornená tmavá krivka, ktorá vyjadruje závislosť medzi množstvom spoločných priateľov a pravdepodobnosťou vzniku prepojenia. Vidíme, že krivka začína výrazne rásť už od *k* rovnajúceho sa 2 (dvaja spoloční priatelia).



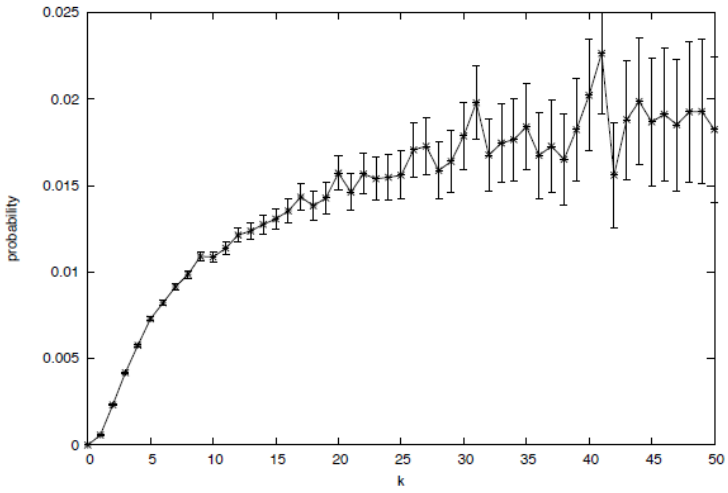
Obrázok 41. Pravdepodobnosť vytvorenia prepojenia (os y) v závislosti od počtu spoločných priateľov (os x) [5].

V tejto práci opisujú aj pravdepodobnosť vytvorenia prepojenia v závislosti od množstva spoločných činností, do ktorých sa ľudia zapoja (ústredný uzáver). V tomto experimente išlo o spoločnú príslušnosť študentov do kurzov. Na obrázku 42 krivka zobrazuje závislosť medzi množstvom spoločných činností a pravdepodobnosťou vzniku prepojenia. Vidíme, že po tri spoločné činnosti pravdepodobnosť výrazne stúpa, potom už zväčšujúci sa počet spoločných činností nemá veľký vplyv.



Obrázok 42. Pravdepodobnosť vytvorenia prepojenia (os y) v závislosti od počtu spoločných záujmov (os x) [5].

Práca Backstroma [6] sa zaoberala vlastnosťami členského uzáveru. Na blogovacej sieti, kde sa používatelia môžu spojiť a kde používatelia môžu patriť do nejakej skupiny, vyhodnotili, ako závisí pravdepodobnosť pripojenia sa do skupiny od množstva priateľov, ktorí tak už urobili. Túto závislosť znázorňuje graf na obrázku 43. Najväčší efekt nastáva pri nižšom počte priateľov zapojených do skupiny, ďalej je menej výrazný, avšak stále významný nárast.



Obrázok 43. Pravdepodobnosť pripojenia sa do skupiny (os y) v závislosti od množstva spoločných priateľov (os x) [6].

10.4 Zhrnutie

Populárne webové aplikácie poskytujúce služby sociálnych sietí podnecujú skúmanie štruktúr sociálnych sietí a interakciu jedincov v sieťach. Jeden z princípov, ktorý na štruktúru zásadne vplýva, je homofília. Medzi podobnými ľuďmi sa vytvárajú prepojenia s veľkou pravdepodobnosťou a naopak ľudia, medzi ktorými sú prepojenia, sa ovplyvňujú svojimi záujmami. Za tieto javy sú zodpovedné mechanizmy selekcie a sociálneho vplyvu. Sociálno-afiliačná sieť berie do

úvahy aj činnosti či záujmy účastníkov v sieti. Ak sa dvaja účastníci zúčastňujú na rovnakej činnosti, vytvára sa medzi nimi prepojenie s veľkou pravdepodobnosťou. Podobne aj činnosť začne byť zaujímavá pre človeka, ktorého kamarát sa o ňu zaujíma. Takéto poznatky z oblasti sociálnych sietí majú veľké uplatnenie pri vyhľadávaní, odporúčaní, či zefektívňovaní marketingu.

Literatúra

[1] Easly D., Kleinberg J.: Networks, *Crowds and Market: Reasoning About a Highly Connected World*. Cambridge, (2010). ISBN: 9780521195331. Dostupné na: http://www.cs.cornell.edu/home/kleinber/networks-book/

[2] McPhearson M. et al.: Birds of a Feather: Homophily in Social Networks. In *Annual Review of Sociology*, (2001).

[3] Crandall, D.: Feedback effects between similarity and social influence in online communities. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08)*, (2008), pp. 160-168.

[4] Christakis N, Fowler J.: The Spread of Obesity in a Large Social Network over 32 Years. In *New England Journal of Medicine*, (2007), pp. 370-379.

[5] Kossinets G., Watts D.: Empirical Analysis of an Evolving Social Network, In *Science 6*, (2006), pp. 88-90.

[6] Backstrom L.: Group Formation in Large Social Networks: Membership, Growth, and Evolution? In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '06)*, (2006), pp. 44-54.

[7] Moody J.: Race, school integration, and friendship segregation in America. In *American Journal of Sociology*, (2001), pp. 679-716.

[8] Pelechrinis K., Krishnamurthy P.: Location Affiliation Networks: Bonding Social and Spatial Information. In *ECML/PKDD: European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, (2012).

[9] Kossinets G.: Origins of Homophily in an Evolving Social Network. In *American Journal of Sociology*, vol. 115, (2009).

[10] Brandes U. et al.: Studying Social Networks: *A Guide to Empirical Research.* Campus Velag, (2012), ISBN: 3593397633.

[11] Wassserman S.: Social Network Analysis: *Methods and Applications.* Cambridge University Press, (1994), ISBN: 0521387078.

[12] Social Networking Statistics. (2014). Dostupné na: http://www.statisticbrain.com/social-networking-statistics/.

[13] Boyd D., Ellison J.: Social network sites: Definition, history, and scholarship. In *Journal of Computer-Mediated Communication*, (2007).

[14] +1's: the right recommendations right when you want them—in your search results. (2011). Dostupné na: http://googleblog.blogspot.sk/2011/03/1s-right-recommendations-right-when-you.html

[15] Domingos, P.: Mining social networks for viral marketing. In *IEEE Intelligent Systems*, (2005), pp. 80-82.

[16] Truthy: Information Diffusion in Online Social Networks. (2013). Dostupné na: http://cnets.indiana.edu/groups/nan/truthy/

[17] Bond, R. M., Fariss, C. J., Jones, J. J., Kramer, A. D., Marlow, C., Settle, J. E., Fowler, J. H.: A 61-million-person experiment in social influence and political mobilization. In *Nature*, (2012), pp. 295-298.

[18] Vojtek, P.: Contribution to Relational Classification with Homophily Assumption. Information Sciences and Technologies. In *Bulletin of the ACM Slovakia*, vol. 2, no. 1 (2010), pp. 26-33.

[19] Michalco, J., Návrat, P.: Arrangement of Face-to-Face Meetings using Social Media. In *Studies in Informatics and Control*. vol. 21, no. 4 (2012), pp. 383-392.

[20] Žilinčík, M., Návrat, P. Kosková, G.: Exploratory search on Twitter utilizing user feedback and multi-perspective microblog analysis. In *PLOS One*. vol. 8, issue 11, (2013), pp. 9.

[21] Korenek, P., Šimko, M.: Sentiment analysis on microblog utilizing appraisal theory. In *World Wide Web: Springer Science-Business Media*, vol. 17, issue. 4, (2014), pp. 847-86.

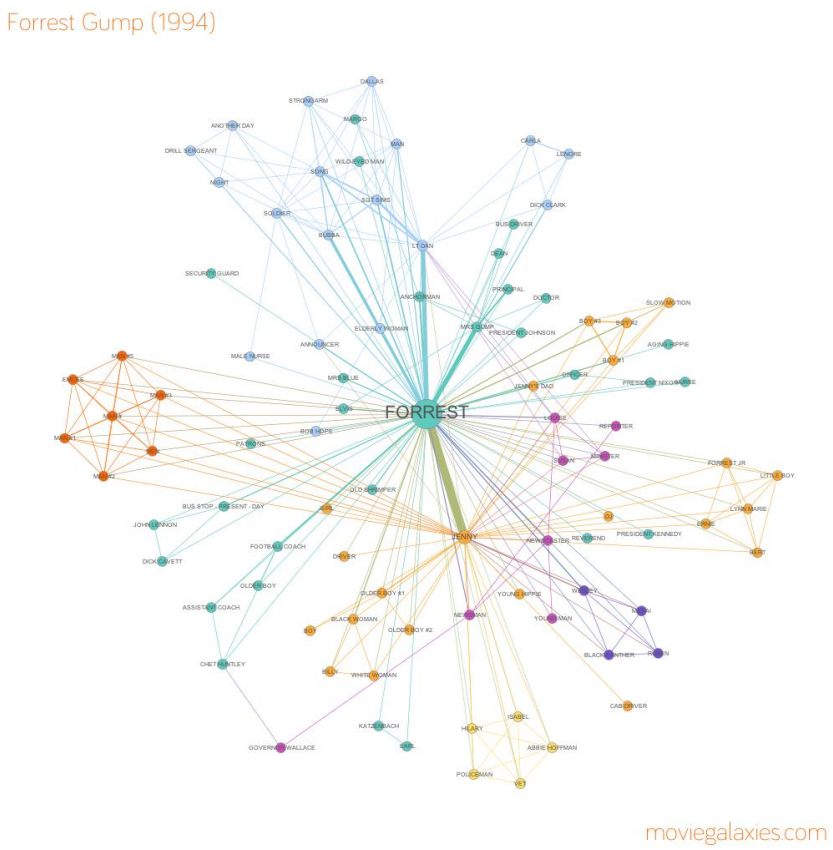
11 Vizualizácia sociálnych sietí

V čase, keď vo veľkom využívame služby sociálnych sietí, na webe vznikajú množstvá sociálnych sietí. Z týchto údajov môžeme vyťažiť nové informácie o vzťahoch medzi členmi siete, ich správaní sa alebo ich vlastnostiach. Takéto informácie sú využiteľné v mnohých odvetviach výskumu aj praxe. Vizualizácia sociálnej siete je prostriedok, ktorý nám uľahčuje analýzu sociálnych sietí.

Vizualizácia sociálnych sietí nám pomáha v jej pochopení. Analýza sociálnej siete je odvetvie sociológie, ktoré sa zaoberá kvantitatívnym ohodnotením roly jedinca v skupine alebo komunite analyzovaním jeho vzťahov s ostatnými členmi siete. Zakladateľ tohto odvetvia je Jacob L. Moreno (1889-1974), ktorý navrhol prvé vizualizácie sociálnych sietí a graf slúžiaci na tento účel – sociogram. Dnes je vizualizácia sociálnej siete bežne používaným prostriedkom pri analýze a prezentácii údajov zo sociálnych sietí. Príkladom môžu byť sociálne siete vytvorené v službách akými sú Facebook, Twitter, LinkedIn. Z tohto dôvodu vzniklo aj veľa vizualizačných nástrojov, ktoré ponúkajú široké možnosti vizualizácie a uľahčujú prácu pri analýze. V tejto kapitole spomíname pár praktických príkladov využitia vizualizácií sociálnych sietí. Ďalej opisujeme jednotlivé prvky sociogramu a jeho vývoj. Na záver uvádzame prehľad najčastejšie používaných vizualizačných nástrojov.

11.1 Príklady vizualizácií sociálnych sietí

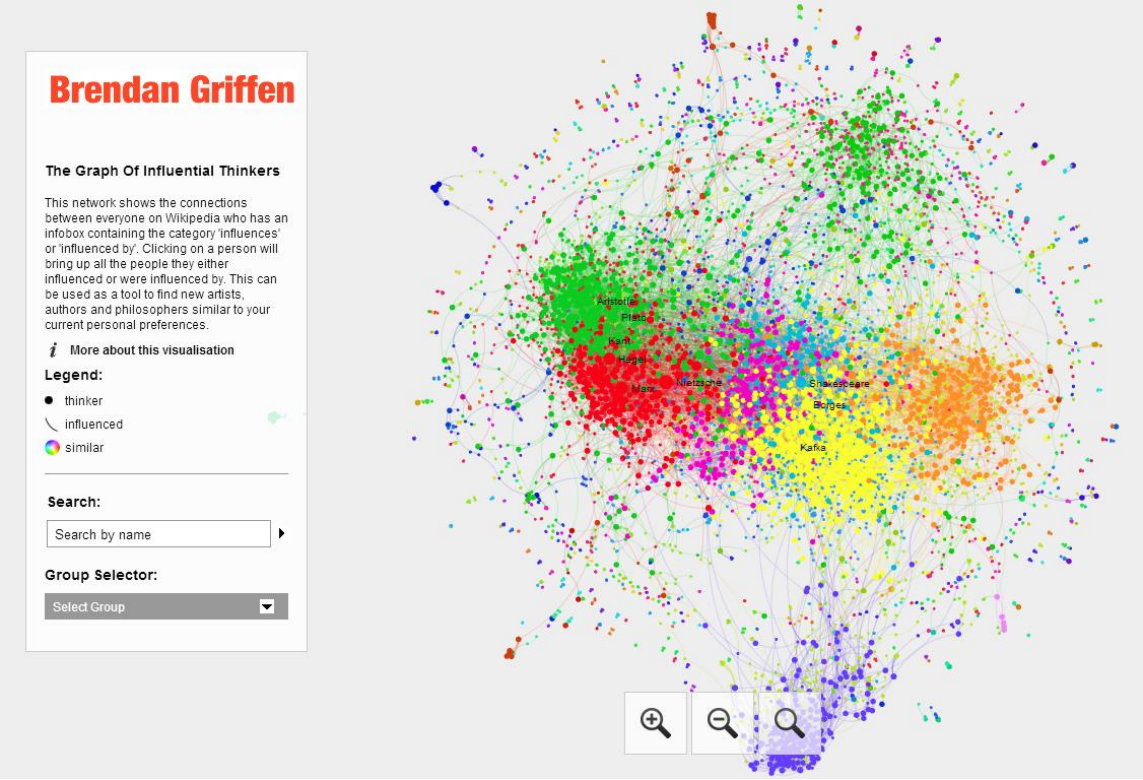
Projekt *moviegalaxies* [1] vznikol spoluprácou výskumníkov z MIT a Kolínskej univerzity. Skúma sociálne siete vo filmoch a ponúka vizuálizácie sietí zo širokého spektra filmov. Vo vizualizácii sa znázorňujú postavy filmu a vzťahy medzi nimi (pozri obrázok 44). Veľkosť uzla vyjadruje dôležitosť postavy v príbehu. Hrúbky hrán medzi uzlami vyjadrujú silu vzťahu medzi postavami. Farebne sú oddelené zhluky postáv, ktoré tvoria určitú dejovú líniu v príbehu. Predmetom výskumu je vnímanie filmu divákmi v závislosti od sociálnej siete vo filme a hľadanie „receptu“ na úspešný film analyzovaním sociálnych sietí.



Obrázok 44. Vizualizácia sociálnej siete filmu Forrest Gump (1994) z projektu *moviegalaxies* [1].

Ďalší projekt [2] z MIT skúma, ako sa ovplyvňujú všetky osobnosti na Wikipédii, napr. filozofi, herci, spisovatelia. Wikipédia pri každej osobnosti uvádza vzťahy: kým bola daná osobnosť ovplyvnená (vzťah *influenced by*) a koho ovplyvnila (vzťah *influenced*). Údaje pre túto vizualizáciu získali z DBpedie. Vo výslednej interaktívnej vizualizácii (pozri obrázok 45) osobnosti znázornili uzlami, ktoré prepojili na základe uvedených vzťahov. Veľkosť uzla zodpovedá množstvu vzťahov – čím väčší uzol, tým väčší vplyv mala daná osobnosť v histórii. Uzly farebne zoskupili do zhlukov, ktoré môžu reprezentovať filozofické smery a zoskupovať podobné osobnosti. Takáto vizualizácia umožňuje napríklad objavovať nové osobnosti a diela, ktoré by sa nám mohli páčiť, na základe podobnosti s našimi obľúbencami.

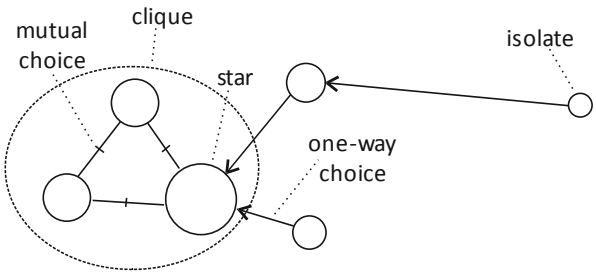
Projekt OrgOrgChart (Organic Organization Chart) [3] sa zameriava na sledovanie vzťahov na pracovisku a organizačnú štruktúru v čase. Počas štyroch rokov zaznamenávali udalosti v zamestnaneckej štruktúre spoločnosti Autodesk research. Časom sa ku spoločnosti pridávali noví zamestnanci, starí odchádzali alebo sa menili riaditelia jednotlivých oddelení. Každý deň zobrazili zamestnaneckú štruktúru v tvare stromu. Jednotlivé obrázky potom pospájali do videa, na ktorom je možné pozorovať zmeny v čase.



Obrázok 45. Vizualizácia osobností z projektu Influential Thinkers [2].

11.2 Sociogram

Sociogram (pozri obrázok 46) vymyslel Jacob L. Moreno [4] na analýzu preferencií v skupine. Dokáže zachytiť štruktúru a vzory v interakciách v skupine. Členovia skupiny sú na sociograme zachytení ako uzly. Preferencie členov skupiny sa označujú orientovanými šípkami. Ak sú preferencie vzájomné, tak sa označujú čiarou pretnutou v strede. Negatívne preferencie sa označujú prerušovanou čiarou.

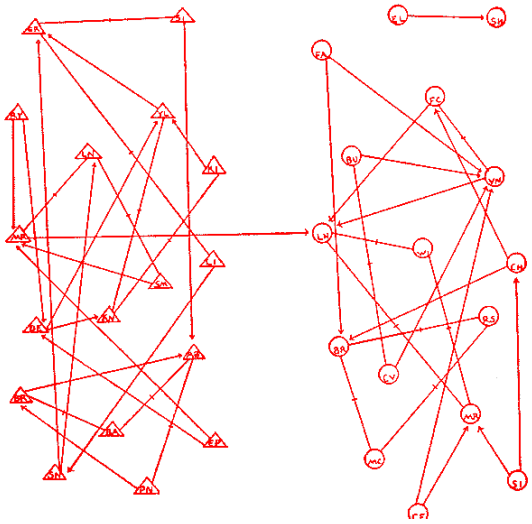


Obrázok 46. Príklad sociogramu.

Na sociograme sa môžu vyskytnúť rôzne kombinácie vzťahov medzi uzlami, tzv. sociometrické vzorce. Dva uzly so vzájomnými preferenciami sa nazývajú *pár*, tri uzly *trojuholník*, viac uzlov s kruhovými preferenciami vytvára *reťazce*. Uzol, do ktorého smeruje veľa preferencií, sa nazýva *hviezda*. Skupina troch a viac uzlov so vzájomnými preferenciami sa nazýva *klika* alebo *partia*,

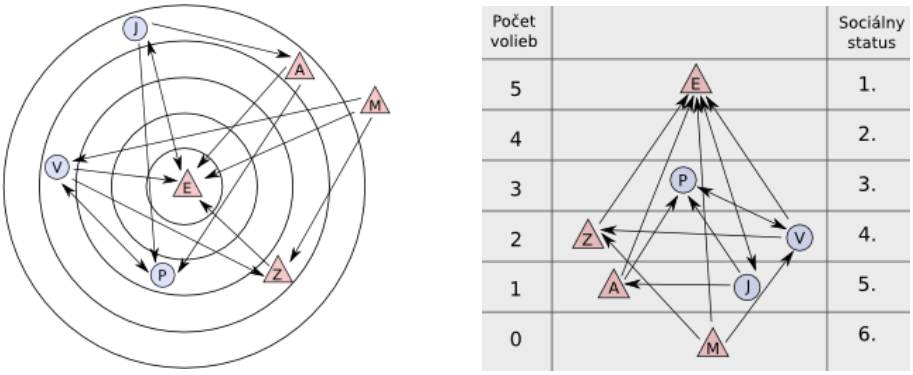
naopak uzol bez preferencií sa nazýva *izolovaný*. Ak má *izolovaný* uzol len vzájomnú preferenciu s *hviezdou*, nazýva sa *šedá eminencia*. Uzol, do ktorého smerujú len negatívne preferencie sa nazýva *odmietaný*. Uzol, z ktorého vychádzajú preferencie, ale žiadne nesmerujú doň je *zabudnutý*.

Uzly môžu mať rôzne podoby. Napríklad ich veľkosť môže závisieť od ich dôležitosti v diagrame. Alebo sa môžu použiť rôzne tvary na odlíšenie skupín uzlov. Moreno vo svojom prvom sociograme použil trojuholník na označenie mužov a kruh pre označenie žien (pozri obrázok 47). V moderných vizualizáciách sociálnych sietí sa sila preferencií často vyjadruje hrúbkou hrany medzi uzlami.



Obrázok 47. Sociogram priateľstiev žiakov základnej školy (trojuholníky chlapci, kruhy dievčatá) [4].

Existuje viacero typov sociogramov. V *klasickom sociograme* je snaha umiestniť najobľúbenejších členov do stredu a izolovaných na okraje. Sociogram sa kreslí tak, aby sa pretínalo čo najmenej hrán. Sila vzťahov medzi jedincami sa vyjadruje vzájomnou vzdialenosťou uzlov. Druhý typ sociogramu je *kruhový sociogram*. Uzly sa umiestňujú na sústredné kružnice. Každá kružnica reprezentuje interval prijatých preferencií. V strede sú umiestnené hviezdy, na okrajoch izolované uzly. V *tabuľkovom sociograme* sa uzly umiestňujú do tabuľky. Tabuľka má toľko riadkov, koľko preferencií má najobľúbenejší uzol. Od prvého riadku sa umiestňujú uzly od najobľúbenejších po menej obľúbené nižšie spolu s napísaným počtom preferencií (pozri obrázok 48).



Obrázok 48. Kruhový (vľavo) a tabuľkový (vpravo) sociogram 10.

11.3 Proces tvorby vizualizácie sociálnej siete

Dnes sa sociogram zostavuje pomocou dostupných vizualizačných nástrojov, ktoré urýchlia proces jeho tvorby a zabezpečia jeho čo najlepší vzhľad, t. j. aby sa pretínalo čo najmenej hrán, mal správne rozloženie, ale aj jeho vzhľad z estetického hľadiska. Proces vytvorenia vizualizácie v nástroji sa skladá z týchto krokov:

1. získanie údajov – zozbieranie údajov, ktoré chceme zobrazovať a analyzovať,
2. výber a úprava údajov do vhodného formátu – každý nástroj podporuje iný vstupný formát údajov, napr. JSON, CSV a pod.,
3. import údajov do vizualizačného nástroja,
4. nastavovanie – v tomto kroku sa vyberá typ vizualizácie a nastavujú sa rôzne parametre sociogramu ako jeho typ, algoritmus na výpočet polohy uzlov, nastavenia hrán a uzlov (farby, tvary a pod.),
5. konečná vizualizácia – výsledok vizualizácie, ktorá môže byť interaktívna alebo len zachytený obraz sociogramu.

11.4 Nástroje na vizualizáciu sociálnych sietí

V závislosti od možnosti využitia nástrojov na vizualizáciu sme ich rozdelili do troch skupín a uvádzame najpopulárnejšie z nich. Do prvej skupiny spadajú nástroje, v ktorých samotných dokážeme vykonať celý proces tvorby vizualizácie – od importu údajov až po prehliadanie konečnej vizualizácie. Druhá skupina nástrojov obsahuje javascriptové knižnice, vďaka ktorým je možné vytvoriť a prehliadať vizualizácie vo webovom prehliadači. Do poslednej skupiny sme zaradili knižnice na vizualizáciu, ktoré možno využiť pri programovaní v rôznych jazykoch.

11.4.1 Stand-alone softvér

Najpoužívanejším nástrojom na vizualizáciu grafov je balík open-source nástrojov *Graphviz*[29] (Graph Visualization Sotware). Vyvíja ho AT&T Labs Research. Skladá sa z viacerých častí.

[29] http://www.graphviz.org/

Jednou z nich je jazyk pre opis grafu DOT, ktorý sa stal bežne používaným formátom pri práci s grafovými údajmi. Nástroj obsahuje príkazový riadok na tvorbu grafov (napr. do pdf, svg výstupu). Ostatné jeho časti sú najmä algoritmy na vykresľovanie rôznych rozložení uzlov v grafe.

Nástroj *Gephi*[30] je ďalším open-source softvérom. Neslúži len na vizualizáciu, ale obsahuje aj analytické funkcie. Naprogramovali ho v jave a vybudovali na platforme NetBeans. Jeho vývoj inicializovali študenti z francúzskej UTC. Použitie nachádza v oblastiach ako sú prieskumná analýza údajov, analýza prepojení, analýza sociálnych či biologických sietí. Existuje doň veľa doplnkov, ktoré vyvíja široká verejnosť. Podporuje veľa formátov vstupných aj výstupných údajov.

V prostredí tabuľkového editora Excel je možné vizualizovať sociálne siete pomocou doplnku *NodeXL*[31](Network Overview, Discovery and Exploration for Excel). Je to open-source softvér, ktorý poskytuje funkcionality pre vizualizáciu aj analýzu údajov. Je vhodný pre používateľov, ktorí nemajú skúsenosti s programovaním.

11.4.2 Vizualizácie vo webovom prehliadači

Najväčšou javascriptovou knižnicou pre vizualizáciu údajov vo webovom prehliadači je *D3.js*[32] (Data-Driven Documents). Využíva technológie CSS a SVG. Je vhodná na efektívne vykresľovanie aj veľkých súborov údajov. D3 podporuje veľká komunita používateľov, ktorí vyvinuli doň rôzne doplnky a vytvorili fórum plné informácií aj pre menej skúsených používateľov.

JavaScript Infovis Toolkit[33] (JIT) poskytuje viaceré formy vizualizácie údajov. Jeho vývoj začal jediný autor, neskôr knižnicu prevzala organizácia Sencha Labs. Knižnica momentálne využíva technológie WebGL a CSS3. Bohužiaľ nie je príliš dobre zdokumentovaná.

Treťou populárnou JavaScript knižnicou je *Processing.js*[34]. Je to javascriptové rozšírenie pre vizuálny programovací jazyk processing. Processing obsahuje sadu príkazov na vykresľovanie rôznych objektov. Využíva sa na výučbu programovania vizualizácií. Táto knižnica sa využíva aj v projekte moviegalaxies [1].

11.4.3 Knižnice pre programovacie jazyky

Vizualizačné knižnice existujú aj pre iné programovacie jazyky ako javascript. Umožňujú, aby sa vizualizácie stali súčasťou akéhokoľvek softvéru. Pre programovací jazyk java je to knižnica *JUNG*[35] (Java Universal Network/Graph Framework). Ponúka širokú škálu grafových algoritmov, ale aj algoritmov pre ich zobrazenie a výpočet polohy uzlov v zobrazení.

[30] https://gephi.org/
[31] http://nodexl.codeplex.com/
[32] http://d3js.org/
[33] http://thejit.org/
[34] http://processingjs.org/
[35] http://jung.sourceforge.net/

Programovací jazyk R slúži sa využíva najmä na analýzu údajov, existujú však doň aj balíčky, pomocou ktorých sa dajú zobrazovať grafy a vytvárať vizualizácie (napr. igraph[36], network, RSiena, RGraphviz). RGraphviz sprístupňuje funkcionality softvéru Graphviz, spomínaného v prvej skupine nástrojov na vizualizáciu, v prostredí R.

Pre jazyk python existuje 2D vykresľovacia knižnica Matplotlib[37] a knižnica NetworkX[38] pre manipuláciu s grafmi, ktorá používa na vykresľovanie Matplotlib alebo Graphviz.

11.5 Zhrnutie

Analýza sociálnych sietí pomocou ich vizualizácie vznikla už v 30. rokoch 20. storočia, keď Jacob L. Moreno zaviedol graf na zobrazenie sociálnej siete – sociogram. Graf siete nám umožňuje ľahšie pozorovať a pochopiť vzťahy medzi členmi siete. Dokážeme z neho čítať sociometrické vzorce, napr. páry, hviezda, izolovaní členovia, kliky. V súčasnosti nám kreslenie sociogramov uľahčuje široký výber nástrojov slúžiacich na vizualizáciu. V období, keď sa služby sociálnych sietí hojne využívajú a poskytujú bohaté zdroje údajov, máme možnosť s využitím vizualizačných nástrojov vyťažiť nové informácie a využiť ich pri výskume v rôznych oblastiach.

Literatura

[1] Moviegalaxies. (2013). Dostupné na: http://moviegalaxies.com/

[2] Griffen, B.: Graphs of Wikipedia: Influential Thinkers. (2013). Dostupné na: http://brendangriffen.com/gow-influential-thinkers/

[3] OrgOrgChart: The Evolution of an Organization. (2013). Dostupné na: http://www.autodeskresearch.com/projects/orgorgchart

[4] Moreno, J. L.: Who shall survive?: A new approach to the problem of human interrelations. In *Nervous and Mental Disease Publishing Co*, (1934).

[5] Gavora, P. et al..: *Elektronická učebnica pedagogického výskumu*. Bratislava, Univerzita Komenského, (2010). Dostupné na: http://www.e-metodologia.fedu.uniba.sk

[36] http://igraph.sourceforge.net/
[37] http://matplotlib.org/
[38] http://networkx.github.com/

12 Kolektívna inteligencia a múdrosť davu

V súčasnosti web ponúka široké spektrum aplikácií, ktoré nám umožňujú efektívne vyhľadávať informácie, spolupracovať alebo komunikovať. Veľká časť týchto aplikácií dokáže fungovať predovšetkým vďaka komunite používateľov, ktorí ich aktívne využívajú a tým či už explicitne alebo len implicitne zdieľajú svoje znalosti. Na využitie takto získaných znalostí celej komunity sa používajú modely kolektívnej inteligencie a múdrosti davu. Napriek tomu, že tieto pojmy sa využívajú v informačných technológiách len veľmi krátku dobu, ich úspešnosť sa už preukázala popularitou a úspešnosťou aplikácií postavených na týchto modeloch. Pri skúmaní modelov kolektívnej inteligencie a múdrosti davu však treba vnímať aj ich negatívne dôsledky, medzi ktoré patrí predovšetkým problém so zabezpečením ochrany súkromia používateľov v priestore internetu.

Znalosti majú už dlhodobo dôležitú rolu v rozličných organizáciách a komunitách, pretože správne udržiavané znalosti reprezentujú kritický faktor pre úspešné rozvíjanie a vzájomné konkurovanie si takýchto organizácií [1]. Z tohto dôvodu sú organizácie a komunity motivované, aby venovali potrebnú pozornosť vytváraniu, zdieľaniu a zdokonaľovaniu znalostí. Predovšetkým proces zdieľania znalostí, počas ktorého si členovia príslušnej komunity navzájom vymieňajú znalosti, sa významne zdokonalil a zefektívnil s nástupom informačných a komunikačných technológií. Významným míľnikom sa stali aplikácie webu 2.0, ktoré priniesli nové možnosti, ako zdieľať znalosti a najmä ako sa aktívne podieľať aj na ich rozširovaní medzi ostatných členov komunity (napr. Wikipédia). Tieto systémy označujeme ako systémy manažmentu znalostí (angl. knowledge management systems).

Teória manažmentu znalostí rozlišuje tri pohľady na znalosti, ktoré sa spracúvajú systémami manažmentu znalostí [12]: znalosť ako objekt, znalosť obsiahnutá v jednotlivcoch

a znalosť obsiahnutá v komunitách. V rámci tohto príspevku sa zameriavame na pohľad znalostí obsiahnutých v komunitách, ktoré sa v teórii manažmentu znalostí nazývajú aj komunity skúseností (angl. communities of practice) [13]. Základným procesom komunít skúseností je kolaboratívne zdieľanie znalostí.

12.1 Modely kolaboratívneho zdieľania znalostí

V súčasnosti sa stalo kolaboratívne zdieľanie znalostí v komunitách ľudí predmetom viacerých výskumných oblastí. Okrem informačných technológií skúmajú túto doménu aj ďalšie disciplíny ako psychológia, sociológia alebo informačné vedy. Výskumné aktivity spadajúce do oblasti informačných technológií vnímajú proces kolaboratívneho zdieľania znalostí predovšetkým z pohľadu modelu kolektívnej inteligencie (angl. collective intelligence).

12.1.1 Kolektívna inteligencia

Definícia. Pre pojem kolektívnej inteligencie neexistuje ustálená definícia. Wikipédia, ako typický príklad systému využívajúceho kolektívnu inteligenciu, definuje tento pojem ako „zdieľanú alebo skupinovú inteligenciu, ktorá vzniká ako dôsledok spolupráce a súťaženia viacerých jednotlivcov a objavuje sa počas procesu hľadania konsenzu“. Pojem kolektívnej inteligencie definuje mierne odlišne Thomas W. Malone, riaditeľ Centra pre kolektívnu inteligenciu, ktoré je súčasťou MIT. Kolektívnou inteligenciou označuje samotné „skupiny jednotlivcov spolupracujúcich na úlohách, ktoré sa zdajú byť inteligentné“ [5]. Spoločnou črtou všetkých definícií je myšlienka, že žiadny z členov príslušnej komunity nevie všetko, ale každý vie niečo a vhodným spojením znalostí jednotlivcov vieme získať rozsiahlu kolektívnu inteligenciu.

Z pohľadu oboch definícií je evidentné, že kolektívna inteligencia existovala už dávno pred vznikom informačných technológií. Typickými príkladmi komunít, ktoré už dlhodobo využívajú princípy kolektívnej inteligencie, sú napr. rodiny, národy alebo armády. Navyše kolektívna inteligencia nie je typická len pre ľudí, ale vyskytuje sa aj u zvierat žijúcich v inteligentných spoločenstvách ako sú napr. včely alebo mravce.

Informačné technológie, predovšetkým internet, však umožnili vznik úplne nových foriem kolektívnej inteligencie. Typickými príkladmi systémov založených na kolektívnej inteligencii je Wikipédia alebo vyhľadávacie nástroje Google. Informačné technológie v takýchto systémoch hrajú kritickú úlohu, pretože s ich využitím dokážu komunity ľudí kolektívne konať oveľa inteligentnejšie, ako to bolo kedykoľvek predtým [5].

Princípy kolektívnej inteligencie. Pierre Lévy, autor knihy *Kolektívna inteligencia: Vznikajúci svet ľudí vo virtuálnom priestore* (angl. Collective intelligence: Mankind's Emerging World in Cyberspace*)*, zadefinoval štyri princípy, ktoré umožňujú existenciu úspešnej kolektívnej inteligencie:

1. **otvorenosť** v zdieľaní znalostí, názorov, atď.

2. **rovnocennosť** členov skupiny alebo komunity, v ktorej neexistuje žiadna formálna hierarchia.
3. **zdieľanie** znalostí, ktoré urýchľuje rozvoj v danej oblasti.
4. **globalizácia**, ktorú zabezpečuje predovšetkým rozvoj informačných technológií.

Obmedzenia. Popri mnohých pozitívach má kolektívna inteligencia aj nezanedbateľné obmedzenia. Ako zdôrazňuje Malone [5], rovnako, ako môže existovať kolektívna inteligencia, tak môže existovať aj kolektívna hlúposť, ak ľudia slepo nasledujú správanie ostatných používateľov alebo ak si navzájom vymieňajú príliš veľa informácií. Ako príklad z oblasti informačných technológií môžeme uviezť automatické dopĺňanie dopytov počas vyhľadávania cez nástroj Google. Na jednej strane je takáto podpora vyhľadávania možná vďaka kolektívnej inteligencii všetkých používateľov, ktorí doteraz vyhľadávali podobné informácie. Na druhej strane, ak všetci používatelia zadávajú do vyhľadávača nesprávne dopyty, budú sa tieto nesprávne dopyty odporúčať ďalším používateľom. Ako reakcia na tento problém vznikol odlišný model kolaboratívneho zdieľania znalostí nazvaný múdrosť davu (angl. wisdom of the crowd).

12.1.2 Múdrosť davu

Definícia. Pojem múdrosti davu zadefinoval James Surowiecki ako „proces, ktorého cieľom je agregovať anonymne vytvorené údaje, pričom hľadáme múdrosť, ktorá vyplýva z odhadov veľkého počtu ľudí, ktorí sa nesmú navzájom ovplyvňovať“ [11]. Základná myšlienka za modelom múdrosti davu tkvie v predpoklade, že keď sú správne zagregované aj nedokonalé hodnotenia jednotlivcov, vieme získať výsledok, ktorý je lepší ako najlepší získaný odhad.

Podobne ako kolektívna inteligencia, aj pojem múdrosti davu je známy už dlhú dobu. Úspešnosť agregovania hodnotenia potvrdil už anglický matematik a štatistik Francis Galton (1822 – 1911), ktorý uskutočnil experiment s odhadom váhy dobytka na statku. Celkovo 787 ľudí dostalo za úlohu odhadnúť váhu dobytka, pričom sa vyskytovali medzi nimi ako experti, tak aj laici. Rovnako sa veľmi odlišovali aj ich odhady. Matematickým priemerom sa získal priemer 1 197 libier, ktorý sa odlišoval len minimálne od skutočnej váhy dobytka, ktorou bolo 1 198 libier. Tento experiment poukázal, že agregované hodnotenie davu bolo presnejšie ako odhad najlepšieho experta, ktoré sa líšilo od skutočnej váhy až o desiatky libier.

Princíp kolektívnej inteligencie si našiel svoje uplatnenie aj v informačných technológiách. V mnohých aplikáciách dokážeme využiť výsledky individuálnej práce jednotlivcov a následne ich vhodne zagregovať. Typickými príkladom takýchto systémov je napr. Mechanický Turek firmy Amazon alebo reCaptcha.

Princípy múdrosti davu. Surowiecki [11] vo svojej knihe *Múdrosť davu* (angl. *Wisdom of the Crowd)* zadefinoval štyri princípy potrebné pre využitie múdrosti davu takto:

1. **rozmanitosť** názorov, aby nebola komunita ľudí rovnorodá.
2. **nezávislosť** hodnotenia jednotlivca, ktoré nesmie byť ovplyvnené hodnotením okolia a každý jednotlivec by mal využiť vlastné znalosti.

3. **decentralizácia**, ktorá zabezpečí, že nikto nediktuje davu svoj názor.
4. **agregácia**, ktorá zahŕňa mechanizmus transformácie hodnotení jednotlivcov do kolektívneho rozhodnutia.

Obmedzenia. Podobne ako model kolektívnej inteligencie, tak aj model múdrosti davu má svoje obmedzenia. V praxi sa používatelia veľmi často navzájom ovplyvňujú a preto nie je možné zabezpečiť splnenie princípu nezávislosti členov komunity. V prípade, že sa členovia komunity navzájom významne ovplyvňujú, môže dôjsť k nežiadúcemu stavu, ktorý sa nazýva skupinové myslenie (angl. groupthink). V takýchto prípadoch chýba v komunite potrebná rôznorodosť a výsledok procesu zdieľania znalostí môže viezť k výrazne odchýleným výsledkom v porovnaní s výsledkami, ktoré by sme dosiahli v dostatočne diverzifikovanej skupine. Ďalším obmedzením modelu múdrosti davu je, že ho je možné použiť len na riešenie objektívnych problémov, kde existuje jednoznačne správna odpoveď.

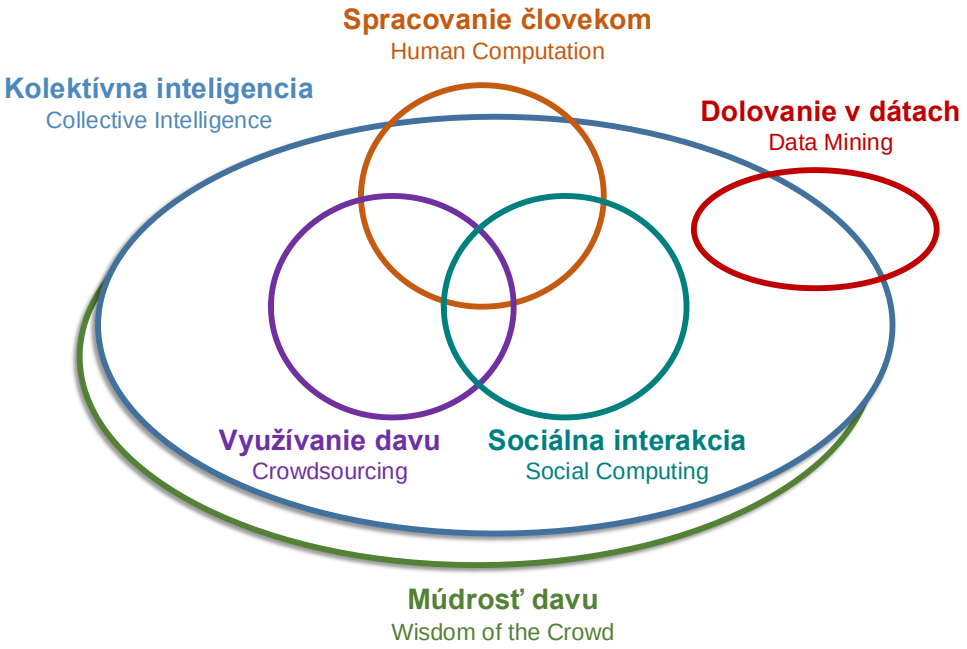
12.1.3 Diskusia

Pojmy kolektívnej inteligencie a múdrosti davu sa v praxi pomerne často zamieňajú a veľké množstvo informačných zdrojov neuvažuje ich principiálne odlišnosti. Jedným z možných dôvodov je, že tieto modely sa navzájom dopĺňajú a často sa v priestore webu využívajú spoločne. Na druhej strane predpokladáme, že poznanie a rozlišovanie ich princípov môže viezť k lepšiemu pochopeniu kolaboratívneho zdieľania znalostí medzi používateľmi, ktoré v konečnom dôsledku vedie k návrhu lepších modelov a metód pre podporu používateľa v informačných systémoch.

12.2 Využitie kolektívnej inteligencie a múdrosti davu na webe

Modely kolektívnej inteligencie a múdrosti davu si našli uplatnenie vo viacerých typoch webových aplikácií. Tieto aplikácie môžeme zaradiť do niekoľkých skupín podľa teórií, na ktorých sú postavené: spracovanie človekom (angl. human computation), využívanie davu (angl. crowdsourcing), sociálna interakcia (angl. social computing) a dolovanie v údajoch (angl. data mining). Všetky tieto teórie pritom stavajú na modeloch kolektívnej inteligencie a múdrosti davu.

V súčasnej literatúre sa nám nepodarilo jednoznačne identifikovať súvislosť medzi analyzovanými modelmi a skupinami aplikácií. Rozhodli sme sa preto znázorniť závislosti medzi týmito pojmami formou diagramu (Obrázok 49), v ktorom oba modely vnímame ako paralelné a navzájom dopĺňajúce sa. Jednotlivé skupiny aplikácií pritom využívajú pojmy jedného alebo oboch podporných modelov.



Obrázok 49. Znázornenie závislostí medzi analyzovanými modelmi a skupinami aplikácií podľa teórií, na ktorých sú postavené (inšpirované [7]).

12.2.1 Spracovanie človekom

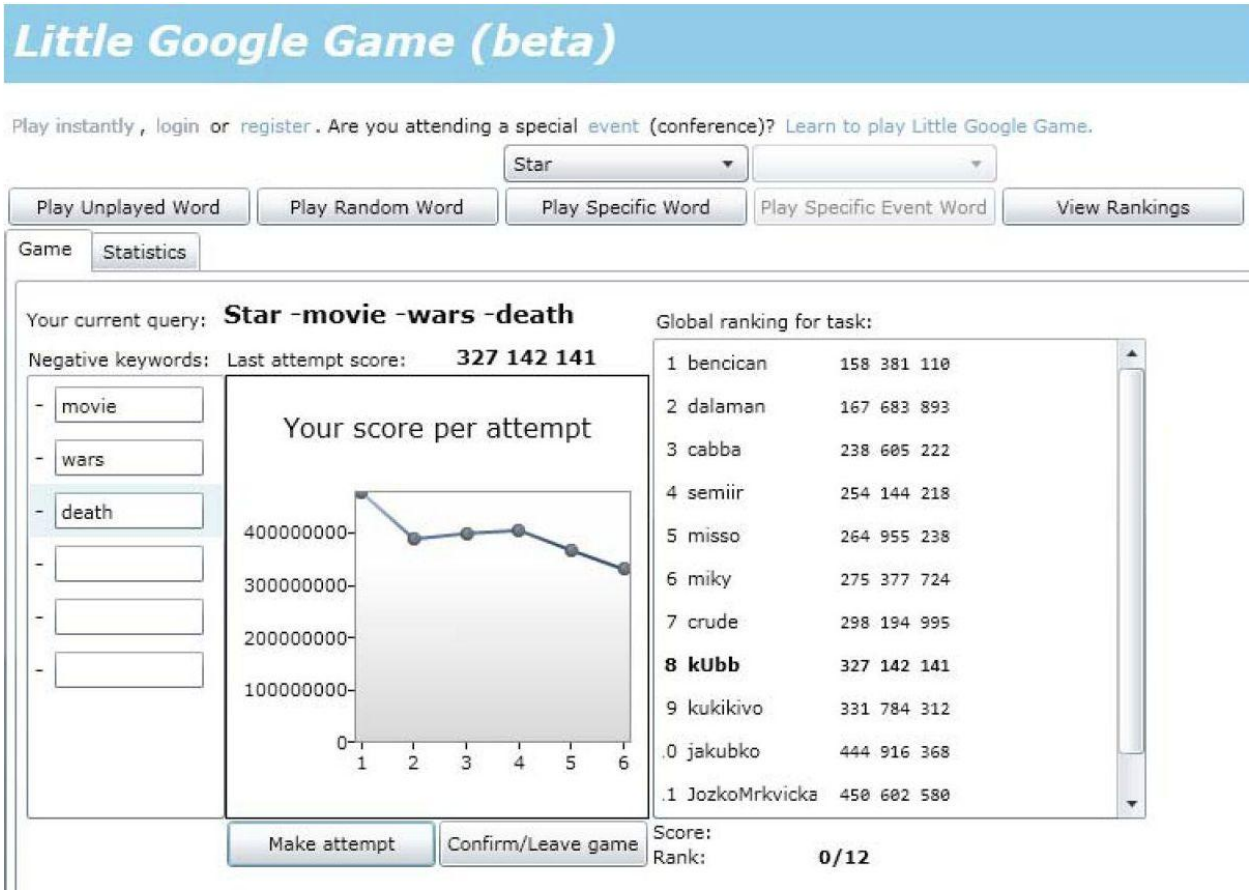
Prvá skupina aplikácií sa zaoberá spracovaním úloh človekom (angl. human computation). Tieto aplikácie sú postavené na paradigme využitia sily ľudského spracovania pre vyriešenie problémov, ktoré nie sú zatiaľ riešiteľné počítačom [1].

Príkladom aplikácií, ktoré využívajú spracovanie človekom, je reCaptcha alebo hry s účelom. reCaptcha je dialógový systém, ktorého primárnym cieľom je zabezpečiť ochranu pred nevyžiadaným obsahom automaticky generovaným škodlivým softvérom. Pre tento účel sa požaduje prepísať z obrázka dvojicu slov, pričom textový prepis jedného zo slov je vopred známy. Prepísaním druhého slova je možné získať jeho dovtedy neznámy textový prepis (Obrázok 50).

Obrázok 50. Ukážka fungovania aplikácie reCaptcha.

Následne viacnásobným overením prepisu toho istého slova je možné zdigitalizovať text, ktorý nie je možný zdigitalizovať štandardnými spôsobmi ako je napr. OCR (angl. optical character recognition). Pomocou systému reCaptcha tak úspešne zdigitalizovali úplný archív novín New York Times. V súčasnosti sa reCaptcha používa aj na rozpoznanie čísel budov z fotografií získaných počas snímkovania ulíc v službe Google Street View.

Ďalším príkladom aplikácií typicky využívajúcich spracovanie človekom sú hry s účelom. Tieto hry kombinujú zábavnú formu hier so získavaním nového obsahu, ktorý nie je možné získať len s využitím počítačov. Hry s účelom sa úspešne použili na získavanie metaúdajov o multimediálnom obsahu (napr. google image labeler) alebo na získavanie vzťahov medzi slovami (napr. little google game - Obrázok 51).

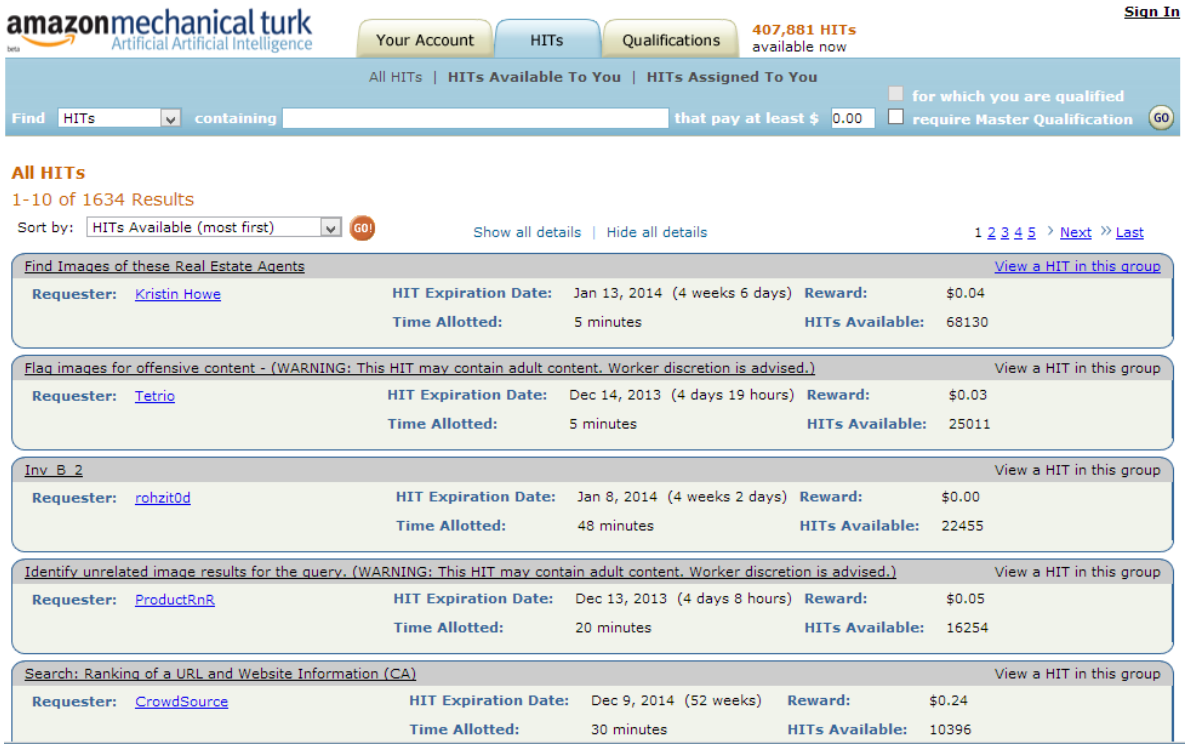


Obrázok 51. Príklad hry s účelom, Little Google Game [10].

12.2.2 Využívanie davu

Zatiaľčo aplikácie využívajúce spracovanie človekom prenášajú časť úlohy z počítača na jednotlivca, aplikácie postavené na využívaní davu prenášajú toto spracovanie ďalej z jednotlivca na celý dav. Teóriu využívania davu môžeme preto definovať ako proces prenesenia úlohy, ktorú tradične vykonáva dedikovaný agent na nedefinovanú, veľkú skupinu ľudí vo forme otvorenej výzvy [4].

V súčasnosti existuje viacero aplikácií, ktoré využívajú dav používateľov na dosiahnutie požadovaných výsledkov. Klasickým príkladom sú aplikácie, ktoré distribujú úlohy (alebo ich časti) zadané jedným používateľom medzi ostatných používateľov, ktorí majú možnosť sa podieľať na ich riešení (väčšinou individuálne). Medzi takéto systémy patrí napr. Mechanický Turek firmy Amazon (Obrázok 52). Poskytuje používateľom rozhranie na výber úloh označovaných ako HIT (angl. human intelligence task), ich vyriešenie a odovzdanie výsledkov autorovi úlohy. Riešiteľom sa zvyčajne následne poskytne finančná odmena.



Obrázok 52. Zoznam dostupných HIT úloh v systéme Mechanický Turek firmy Amazon.

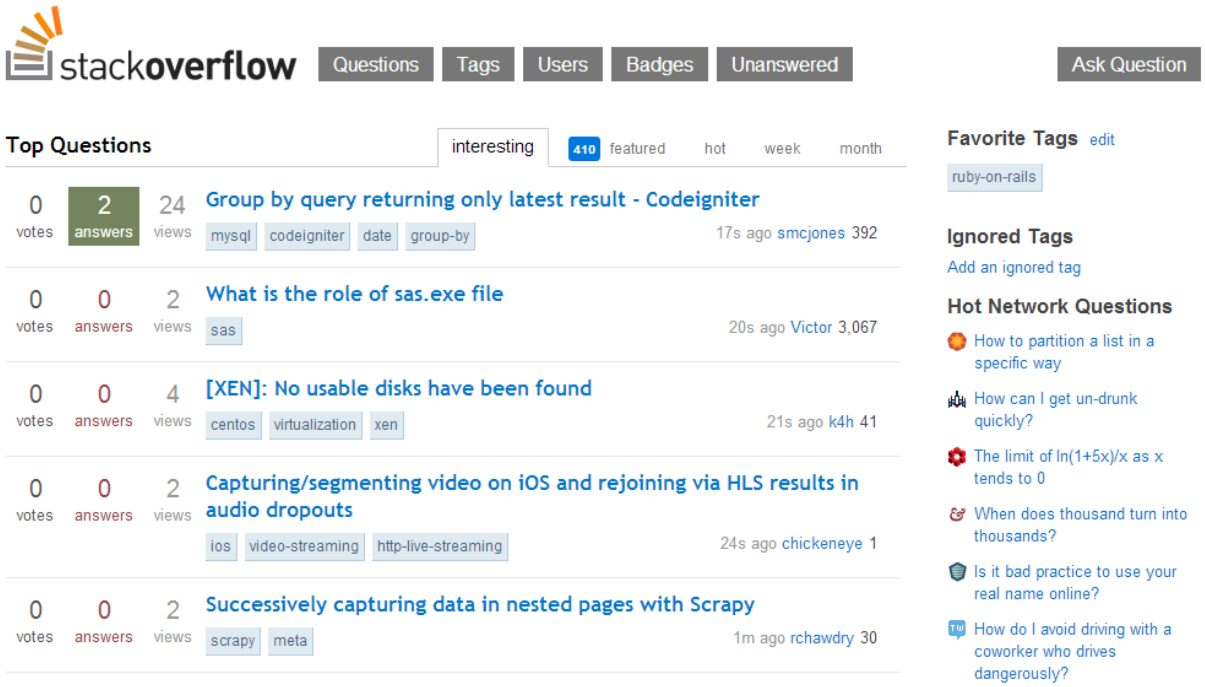
Na využívaní davu sú postavené aj ďalšie aplikácie, ako napr. nástroje pre sociálne značkovanie (angl. social bookmarking), kde je typickým predstaviteľom služba del.icio.us.

12.2.3 Sociálna interakcia

Tretiu skupinu tvoria aplikácie a služby založené na sociálnej interakcii (angl. social computing), ktoré podporujú kolektívne akcie a sociálnu interakciu s významným zapojením multimediálnych informácií a s rozvojom obsiahnutých znalostí [8].

Záber aplikácií spadajúcich do tejto kategórie je veľmi široký. Ich spoločnou črtou je, že vždy zahŕňajú ľudí, ktorí vystupujú v sociálnej roli a následne technológie zabezpečujú komunikáciu medzi nimi [7]. Patria sem predovšetkým webové služby sociálnych sietí (napr. Facebook, Google+), blogy, wiki stránky (napr. Wikipédia) alebo systémy pre odpovedanie otázok v komunitách (napr.Yahoo! Answers).

Predmetom výskumu v rámci dizertačného projektu [9] sú práve systémy pre odpovedanie otázok v komunitách (angl. community question answering - CQA), ktoré poskytujú ľuďom možnosť pýtať sa na rozličné otázky a na druhej strane, poskytovať odpovede na otázky ostatných používateľov. Medzi najpopulárnejšie CQA systémy patria Yahoo! Answers alebo Stack Overflow (pozri Obrázok 53). Tento druh systémov pre zdieľanie znalostí je veľmi úspešný a efektívny predovšetkým v prípadoch, keď nie je možné jednoduchým spôsobom nájsť požadovanú odpoveď pomocou štandardných vyhľadávačov ako napr. Google alebo Bing. CQA systémy už zaujali milióny používateľov a zaznamenali viac ako miliardu poskytnutých odpovedí. Popularita CQA systémov preto dokazuje, že ich koncept založený na kolektívnej inteligencii je úspešný spôsob ako zdieľať svoje znalosti.



Obrázok 53. CQA systém Stack Overflow pre kolaboratívne odpovedanie na otázky položené používateľmi komunity.

12.2.4 Dolovanie v údajoch

V predchádzajúcich prípadoch aplikácie explicitne zasahovali do správania sa používateľov, aby využili výhody modelov kolektívnej inteligencie a múdrosti davu. Rovnako je však možné využívať tieto znalosti implicitne bez aktívneho zasahovania do bežného správania používateľov. Posledná skupina aplikácií preto využíva proces dolovania v údajoch (angl. data mining), ktorý sa definuje ako aplikácia špecifických algoritmov pre objavovanie vzorov v údajov [3].

Vďaka narastajúcemu množstvu údajov na webe sa stalo dolovanie v údajoch bežnou súčasťou mnohých webových aplikácií. Pomocou dolovania údajov dokážeme identifikovať vzory v správaní sa používateľov, pomocou ktorých môžeme v ďalšom kroku podporiť používateľov a to napr. personalizovať systémy, poskytovať odporúčanie atď.

Typickým príkladom aplikovania dolovania údajov za účelom personalizovaného odporúčania je kolaboratívne filtrovanie známe napr. z webovej služby Amazon, kde je pri každom produkte dostupný zoznam produktov, ktoré zákazníci kupujú zároveň s práve zobrazeným produktom.

12.2.5 Diskusia

Podobne ako pri modeloch kolektívnej interakcie a múdrosti davu, tak aj pri analyzovaných skupinách aplikácií nie sú hranice jednoznačne definované a navzájom sa prekrývajú. Existujúce riešenia preto často stavajú na princípoch viacerých teórií a vhodne ich kombinujú. Aby sme dokázali lepšie porozumieť webovým systémom založeným na kolaboratívnom zdieľaní znalostí v komunitách, vypracovali sme prehľad aplikovania analyzovaných teórií a modelov v súčasných webových aplikáciách (Tabuľka 2).

Tabuľka 2. Prehľad aplikovania princípov analyzovaných teórií a modelov v systémoch súčasného webu.

Systém	Využívanie davu	Spracovanie človekom	Sociálna interakcia	Dolovanie v údajoch	Kolektívna inteligencia	Múdrosť davu
reCaptcha	○	●	○	○	○	●
Hry s účelom	●	●	○	○	●	●
HIT systémy (Amazon MTurk)	●	●	○	○	○	●
Sociálne značkovanie (del.icio.us)	●	●	●	○	●	○
Wikipedia	●	○	●	○	●	○
CQA (Stack Overflow)	●	●	●	○	●	●
Sociálne siete (Facebook, blogy)	○	○	●	●	●	●
Google Pagerank	●	○	○	○	○	●
Kolaboratívne filtrovanie	○	○	●	●	○	●

12.3 Zhrnutie

Úspešnosť modelov kolektívnej inteligencie a múdrosti davu a teórií na nich založených dokazujú príklady mnohých populárnych aplikácií, ktoré sú už neodmysliteľnou súčasťou súčasného

webu. Tento aspekt rozpoznala aj komunita výskumníkov, v ktorej sa tieto modely stali predmetom mnohých výskumov. Dokazuje to aj vznik novej konferencie *Collective Intelligence*, ktorá sa konala po prvý raz v roku 2012 a organizujú ju výskumníci z MIT.

Napriek mnohým pozitívam, ktoré nám prináša využitie kolektívnej inteligencie a múdrosti davu na webe, je potrebné si uvedomovať aj negatívne následky týchto pojmov. Kritickým problémom sa stáva napr. ochrana súkromia používateľov na webe. Tento problém spôsobuje chýbajúci etický kódex, ktorý by jednoznačne upravoval, do akej miery je možné využívať informácie a zaznamenané činnosti používateľov na internete. S vďakou uvádzame, že text tejto kapitoly vychádza okrem iných z obsahu prednášky, ktorej autorom je McCrown[6].

Literatúra

[1] Ahn, L.: *Human computation*. PhD. thesis Carnegie Mellon University Pittsburgh, (2005).

[2] Bollinger, A.S., Smith R.D.: Managing Organizational Knowledge as a Strategic Asset. In *Journal of Knowledge Management, vol.* 5, no. 1, (2001), pp. 8–18.

[3] Fayyad, U.: *Knowledge Discovery and Data Mining: Towards a Unifying Framework*. AAAI Press, (1996), pp. 82-88.

[4] Howe, J.: *The Rise of Crowdsourcing*. Wired, (2006).

[5] Malone, T.W.: Collective Intelligence: A Conversation with Thomas W. Malone. (2012). Dostupné na http://edge.org/conversation/collective-intelligence

[6] McCown, F.: *Introduction to Web Science*. Harding University, (2013). Dostupné na http://www.harding.edu/fmccown/classes/comp475-s13

[7] Quinn, A.J., Bederson, B.B.: Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1403-1412.

[8] Parameswaran, M., Whinston, A.B.: Social Computing: An Overview. In C*ommunications of the Association for Information Systems*, vol. 19, article 17, (2007).

[9] Srba, I., Bieliková, M.: Adaptive Support for Educational Question Answering. In *Proceedings of the Doctoral Consortium at the European Conference on Technology Enhanced Learning 2013*, (2013), pp.109–114.

[10] Šimko, J.: Harnessing manpower for creating semantics. In *Information Sciences and Technologies Bulletin of the ACM Slovakia*, vol. 5, issue 3, (2013), pp. 32-40.

[11] Surowiecki, K.: *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday, (2004).

[12] Wasko, M.M., Faraj, S.: It Is What One Does': Why People Participate and Help Others in Electronic Communities of Practice. In *The Journal of Strategic Information Systems*, vol. 9, issue 2-3, (2000), pp. 155–173.

[13] Wenger, E.: *Communities of Practice: Learning, Meaning, and Identity.* Cambridge University Press, (1998).

13 Odporúčacie systémy

V živote sa často riadime radami a odporúčaniami ľudí, ktorých poznáme, a ktorým dôverujeme, či už pri organizácii voľného času (napr. aký film si pozrieť), ale aj pri dôležitejších rozhodnutiach. Preniesť tento koncept do prostredia webu sa snažia odporúčacie systémy. V tejto kapitole analyzujeme rôzne prístupy, spôsoby ich kombinácie, aktuálne výskumné smery, ako aj problémy, ktoré sa s odporúčaním asociujú.

Jedným zo spôsobov, ako sa vysporiadať so zahltením informácií, ktorému musíme na webe čeliť, je *filtrovanie* – obmedzenie informačného priestoru na základe zvolených kritérií, aby sme z množstva dostupných informácií nakoniec museli prijať len tie, ktoré majú pre nás nejakú hodnotu.

Na tomto koncepte sú založené odporúčacie systémy (angl. *recommender systems*), ktorých úlohou je odporučiť nám (vyfiltrovať) informácie (noviny, produkty a pod.), ktoré by nás mali zaujímať (na základe toho, čo o nás odporúčací systém vie, t.j. väčšinou na základe histórie našej interakcie so systémom). Príkladom odporúčacieho systému (jedného z prvých komerčných systémov svojho druhu) je online predajca *Amazon*[39], ktorý je zobrazený na obrázku 54.

13.1 Typy odporúčačov

Existuje viacero typov odporúčacích systémov, t. j. odporúčačov. Podrobný prehľad nájdeme napr. v [7]. Na základe informácií, ktoré sa zohľadňujú pri odporúčaní, rozlišujeme tieto typy:

- kolaboratívne,
- založené na obsahu,
- znalostné,
- sociálne,
- hybridné.

[39] http://www.amazon.com/

Obrázok 54. Príklad odporúčania na Amazone. Používatelia môžu ohodnotiť položku, povedať systému, že ju už vlastnia, alebo ich nezaujíma. Systém sa tiež snaží vysvetliť, prečo si myslí, že by mohla daného používateľa odporučená položka (v tomto prípade kniha) zaujímať.

Kolaboratívnemu odporúčaniu, odporúčaniu založenému na obsahu a hybridnému odporúčaniu sa venujeme v ďalších častiach tejto kapitoly. Sociálne odporúčanie a využitie informácií o kontexte pri odporúčaní analyzujeme v časti 13.2. Znalostné odporúčanie podrobnejšie nerozoberáme, keďže je v porovnaní s predchádzajúcimi prístupmi oveľa menej využívané. Je založené na pravidlách (znalostiach), ktoré musel niekto (väčšinou doménový expert) reprezentovať v odporúčacom systéme. Často majú podobu asociatívnych pravidiel (ak si X kúpil položku A, tak je pravdepodobné, že ho bude zaujímať aj položka B).

Na základe toho, komu odporúčame, rozlišujeme odporúčanie:

- jednotlivcom
- skupinám.

Väčšina príkladov, ktoré v tejto kapitole uvádzame, predpokladá odporúčanie jednotlivcom. Odporúčanie skupinám je stále relatívne málo preskúmaná oblasť, dobrý prehľad je možné nájsť v [2] alebo v [3].

13.1.1 Kolaboratívne odporúčanie

Kolaboratívne odporúčanie (angl. väčšinou označované ako *collaborative filtering – CF*) [9] je založené na predpoklade, že ak sa ľuďom (používateľom) s podobným vkusom (chápaným ako podobná história záujmov) páči X, tak sa bude X páčiť aj inému, im podobnému používateľovi. Snahou je odhadnúť (predpovedať) hodnotenie používateľa pre ešte nevidenú (nehodnotenú) položku na základe hodnotení podobných používateľov. Kľúčové je preto pre tento spôsob odporúčania nájdenie podobných používateľov (s podobným vkusom). Využívajú sa pritom rôzne miery podobnosti:

- kosínusová podobnosť,

- Jaccardov index,
- euklidovská vzdialenosť,
- Pearsonov korelačný koeficient,
- manhattanská vzdialenosť a iné.

Vstupom pre výpočet podobnosti sú väčšinou vektory reprezentujúce používateľov a ich interakciu so systémom – ako hodnotili jednotlivé položky (knihy, filmy, novinové články). Ide teda o formu *explicitnej spätnej väzby* používateľa (používateľ nám explicitne hovorí, či sa mu daná položka páči a ako). Hodnotenia môžu byť intervalové (napr. na intervale od 1 do 5), binárne (páči sa mi to, nepáči sa mi to) alebo unárne (páči sa mi to).

Hlavnou výhodou uvedeného spôsobu odporúčania je jeho nezávislosť od obsahu – môžeme rovnakým spôsobom odporúčať multimédiá, či textové dokumenty, hoci ony samotné sa reprezentujú rozlične.

13.1.2 Odporúčanie založené na obsahu

Odporúčanie založené na obsahu (angl. *content-based recommendation*) [6] vychádza z predpokladu, že sa používateľovi bude páčiť položka X, lebo sa mu predtým páčili položky, ktoré sa jej podobajú. Je možné použiť rovnaké miery podobnosti ako v prípade kolaboratívneho odporúčania (kosínusová podobnosť, Jaccardov index atď.). Hlavný rozdiel je v tom, že nehľadáme podobných používateľov, ale podobné dokumenty (položky).

Je preto dôležité, ako sa jednotlivé položky reprezentujú, čo často závisí od domény – v prípade novinových článkov (alebo vo všeobecnosti textových dokumentov) môžeme použiť reprezentáciu založenú na kľúčových slovách, v ktorej sa dokument reprezentuje ako množina slov (angl. *bag of words*), pričom sa zanedbávajú pokročilejšie syntaktické a sémantické vzťahy. V prípade multimédií (obrázkov, hudby, videí) sa musíme spoľahnúť na dostupné metaúdaje (napr. názov interpreta, názov pesničky, rok vydania a pod.).

Podobne ako pri kolaboratívnom odporúčaní používatelia hodnotia položky v systéme. Okrem toho môžeme využiť aj nepriame indikátory záujmu o obsah, tzv. *implicitnú spätnú väzbu*, ako napr. čas strávený na položke, kliknutie na položku v zozname výsledkov a pod.

Hlavnou výhodou tohto prístupu je (na rozdiel od kolaboratívneho odporúčania) nezávislosť od používateľov.

13.1.3 Porovnanie

Oba uvedené spôsoby, t. j. kolaboratívne odporúčanie aj odporúčanie založené na obsahu majú svoje výhody aj nevýhody. Pri ich porovnaní sa zameriame na tri hľadiská:

- doména
- problém studeného štartu
- škálovateľnosť.

Keďže kolaboratívne odporúčanie závisí od používateľov a ich hodnotení, je vhodné pre domény a systémy, v ktorých máme alebo očakávame viac používateľov ako možných položiek obsahu. Taktiež je vhodné použiť tento prístup, ak máme zložitý obsah, ktorý je náročné reprezentovať. Kolaboratívne odporúčanie tým, že je nezávislé od obsahu, môže odporúčať akýkoľvek obsah. Naopak, odporúčanie založené na obsahu je vhodnejšie v doménach, kde máme málo používateľov (vzhľadom na počet položiek obsahu), pričom obsah je dobre štruktúrovaný, alebo sú k nemu dostupné metaúdaje.

Pri probléme studeného štartu sa rozlišujú dva aspekty:

- *nová položka obsahu v systéme* – pri kolaboratívnom odporúčaní ju nevieme odporučiť, dokým ju niekto neohodnotí; pri odporúčaní založenom na obsahu tento problém nie je
- *nový používateľ* – v oboch prípadoch platí, že nevieme preňho odporúčať, dokým nemá hodnotenia, pretože nemáme ako vypočítať podobnosti; v prípade odporúčaní založenom na obsahu však vieme podobný obsah odporúčať už po prvom hodnotení, pri kolaboratívnom odporúčaní ich potrebujeme zväčša viac (môže byť problém s riedkymi hodnoteniami).

Pri škálovateľnosti musíme zvážiť počet položiek, resp. používateľov v systéme, keďže porovnanie voči všetkým (za účelom zistenia podobnosti) môže byť často neefektívne – namiesto toho môžeme porovnávať podobnosť len voči vzorke, prípadne použiť zhlukovanie na nájdenie najbližších susedov. Pri odporúčaní na základe obsahu treba tiež analyzovať obsah, aj keď toto sa udeje väčšinou len raz vo fáze predspracovania pri pridaní novej položky.

13.1.4 Hybridné odporúčanie

Keďže rôzne prístupy odporúčania majú svoje výhody a svoje nevýhody, často sa kombinujú pre dosiahnutie optimálnych výsledkov. Základné stratégie kombinovania sú [1]:

- *váhovanie* – konečné skóre položky sa počíta ako kombinácia jednotlivých komponentov s rôznymi váhami
- *mix* – odporúčania z rôznych systémov sa prezentujú spolu
- *prepínanie* – systém vyberie pre danú situáciu jeden z odporúčačov (na základe stanovených pravidiel)
- *kaskáda* – odporúčače majú pevne stanovenú prioritu, pričom tie s nižšou prioritou rozhodujú situácie, keď sa odporúčače s vyššou prioritou nevedia rozhodnúť medzi dvomi a viacerými položkami (majú rovnaké skóre)
- *meta-odporúčanie* – výstup odporúčania prvého algoritmu je vstupom pre druhý algoritmus.

13.2 Ďalšie smery výskumu

Jedným z aktuálnych smerov výskumu v oblasti odporúčania sú metódy uvažujúce kontext používateľa. Konceptuálne môžeme rozlíšiť medzi dvomi prístupmi [10]:

- *odporúčanie zohľadňujúce kontext* (angl. *context-aware*) – integruje informácie o kontexte a filtruje na základe nich vhodný obsah
- *odporúčanie založené na kontexte* (angl. *context-based*) – využíva pravidlá založené na kontexte (napr. „*Ak prší, odporuč činnosť v budove.*“); možno ich zaradiť do znalostných prístupov.

Dôležité je tiež zadefinovať, čo všetko považujeme za kontext. Názory na to sa líšia, vo všeobecnosti však môžeme za kontext prehlásiť akúkoľvek informáciu o používateľovi. Najčastejšie typy kontextov sú [8]:

- *kontext úlohy* – čo je cieľom používateľa
- *sociálny kontext* – kto sú priatelia, známi či kolegovia používateľa
- *osobný kontext* – tu môžeme zahrnúť jednak fyziologický kontext (výška, váha, vek, pohlavie a pod.), jednak mentálny (nálada, expertíza, záujmy)
- *časopriestorový kontext* – poloha, smer, čas
- *kontext prostredia* – typ zariadenia, svetelné podmienky a i.

Častým problémom býva chýbajúca alebo nepresná kontextová informácia pre používateľov, v takom prípade sa ju môžeme pokúsiť na základe rôznych heuristík odhadnúť [10].

Ak zohľadníme sociálny kontext, hovoríme o *sociálnom odporúčaní*, čo je ďalšie aktuálne smerovanie najmä s ohľadom na rozmach používania webových služieb sociálnych sietí, ako je *Facebook*, *Twitter* a pod. Sociálne odporúčanie často pracuje s dôverou používateľov; používatelia majú tendenciu odporúčaniu viac dôverovať, ak jeho zdrojom sú ich priatelia, resp. ľudia, ktorých poznajú alebo považujú za autority. Dôležitou je práve identifikácia autorít v sociálnych sieťach, pretože tieto nám pomáhajú pochopiť šírenie informácií alebo trendov medzi používateľmi.

13.3 Problémy súvisiace s odporúčaním

S odporúčaniami sa spája viacero problémov. Na jeden z nich, tzv. problém bubliny (angl. *filter bubble*), poukázal Pariser [5], keď si všimol, že mu Facebook začal postupne skrývať príspevky jeho priateľov, ktorí mali iné politické presvedčenie. Odporúčač ho tak uzatvoril do umelo vytvorenej bubliny – sveta, v ktorom nie je konfrontovaný s názormi, s ktorými nesúhlasí alebo sa mu nepáčia. Pri návrhu algoritmov by sme sa tomu mali vyhnúť napr. tak, že nebudeme odporúčať len položky s najvyšším skóre, ale vždy zahrnieme aj niečo s menším skóre, kde je šanca, že si používateľ takpovediac rozšíri svoje obzory.

Iný problém je *náchylnosť algoritmov odporúčania na manipuláciu*. Keďže sú založené na odporúčaniach používateľov, môže si výrobca nejakého výrobku alebo poskytovateľ nejakej služby zaplatiť ľudí, ktorí budú jeho výrobky a služby vždy hodnotiť pozitívne a naopak výrobky

a služby konkurencie negatívne. Dômyselnejší prístup je hodnotiť podobne ako ostatní používatelia, aby sa zvýšila podobnosť „útočníka“ s ostatnými a systém im podsunul ním želané odporúčania.

S týmto súvisí aj *problém súkromia*, najmä ak sa berie do úvahy aj implicitná spätná väzba, napr. či používateľ čítal alebo nečítal nejaký článok, či pozeral alebo nepozeral nejaké video a toto sa využíva pri kolaboratívnom odporúčaní – môžu sa tak odporučiť aj veci, ktoré sú pre používateľa citlivé. Známym je prípad, keď poskytovateľ internetovej televízie *Netflix* zverejnil anonymizovanú sadu údajov s hodnoteniami používateľov a neskôr sa zistilo, že je možné spätne jednoznačne identifikovať používateľov na základe čiastkovej informácie o nich [4] (napr. že pozerali daný film v nejakom konkrétnom čase).

13.4 Zhrnutie

Odporúčacie systémy predstavujú efektívny spôsob filtrovania informačného priestoru. Existuje viacero spôsobov, ktoré majú vo vzájomnom porovnaní rôzne výhody a nevýhody, je preto často vhodné ich kombinovať. Dôležitý pri odporúčaní je kontext používateľa. Viac úsilia by sa však pri návrhu systémov malo venovať nielen zvyšovaniu ich presnosti, ale aj odolnosti voči manipulácii a zneužitiu súkromia.

Literatúra

[1] Burke, R.: Hybrid web recommender systems. In *The adaptive web, LNCS 4321*, (2007), pp. 377–408.

[2] Jameson, A., Smyth, B.: Recommendation to groups. In *The adaptive web, LNCS 4321*, (2007), pp. 596–627.

[3] Kompan, M.: Group and single-user influence modeling for personalized recommendation. In *Information Sciences and Technologies Bulletin of the ACM Slovakia*, (2014).

[4] Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In *IEEE Symposium on Security and Privacy - SP '08*, (2008). pp. 111–125.

[5] Pariser, E.: *The filter bubble: How the new personalized web is changing what we read and how we think*. Penguin Press, (2011). ISBN 978-0-143-12123-7.

[6] Pazzani, M., Billsus, D.: Content-based recommendation systems. In *The adaptive web, LNCS 4321*, (2007), pp. 325–341.

[7] Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: *Recommender systems handbook*. Springer, (2011). ISBN 978-0-387-85819-7.

[8] Ruthven, I.: Information retrieval in context. In *Advanced Topics in Information Retrieval*, (2011), pp. 187–207.

[9] Schafer, J. Ben et al.: Collaborative filtering recommender systems. In *The adaptive web, LNCS 4321*, (2007), pp. 291–324.

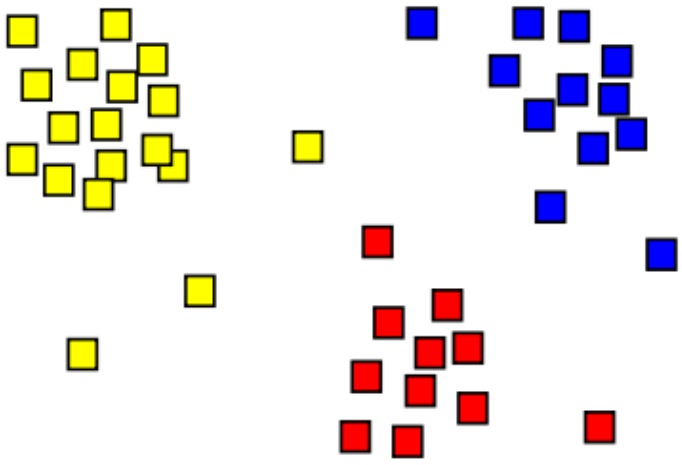
[10] Zeleník, D.: Reducing the sparsity of contextual information for recommendation. In *Information Sciences and Technologies Bulletin of the ACM Slovakia*, (2014).

[11] Andrejko, A., Bieliková, M.: Comparing Instances of Ontological Concepts for Personalized Recommendation in Large Information Spaces. In *Computing and Informatics* vol. 28, no. 4, (2009), pp. 429-452

14 Zhlukovacie algoritmy

S generovaním veľkého množstva údajov vďaka prístupnosti informačných a komunikačných technológií sa objavujú nové výzvy pri ich spracúvaní. V už nazhromaždených údajoch je možné nachádzať nielen už známe informácie, ale aj nové poznatky. Pomocou algoritmov sme napríklad schopní s údajmi pracovať - objavovať a vizualizovať skupiny (zhluky) vecí, ktoré sú nejakým spôsobom prepojené.

S údajmi je možné pracovať rôznymi spôsobmi. Jedným z takýchto spôsobov je napríklad aj zhlukovanie údajov. Zhlukovanie je rýchlo sa rozvíjajúcou oblasťou. Na jeho rozvíjaní sa podieľa výskum v oblasti analýzy údajov, štatistiky, strojového učenia, biológie či marketingu. Zhlukovanie údajov má za úlohu nachádzať údaje, ktoré sú nejakým spôsobom navzájom prepojené. Môže ísť napríklad o nachádzanie zákazníkov s podobnými nákupnými zvykmi, hľadanie stránok s podobnou tématikou alebo detekciu skupín génov, ktoré vykazujú podobné správanie. Zhlukovanie údajov sa tiež zaoberá spôsobmi ich vizualizácie (Obrázok 55).



Obrázok 55. Vizualizácia zhlukov [1].

14.1 Ohodnotenie údajov

Na to, aby bolo možné údaje nej akým spôsobom rozčleniť do zhlukov, treba tieto zhlukované položky nejakým spôsobom ohodnotiť - určiť im číselné skóre, ktoré ich "opisuje". Napríklad:

- zákazníkov možno opísať množstvom nákupov za mesiac
- filmy možno opísať ohodnotením od kritikov
- dokumenty možno ohodnotiť počtom použitých určitých slov.

Na ohodnotenie údajov môžeme aplikovať množstvo vzťahov, ktoré nám kvantitatívne ohodnotia podobnosť resp. vzdialenosť údajov. Vo všeobecnosti platí, že čím sú dva objekty od seba vzdialenejšie, tým sú si menej podobné. Na ohodnocovanie môžeme použiť napríklad:

- euklidovskú vzdialenosť
- manhattanovskú vzdialenosť
- korelačný koeficient (Pearson's r)
- kosínusovú podobnosť
- Jaccardov koeficient
- a iné...

Existuje mnoho metód použiteľných na zhlukovanie údajov. V tomto texte si bližšie opíšeme dve: metódu hierarchického zhlukovania a metódu *K*-priemerov (anglicky *K*-means, slovensky tiež *K*-jadier) zhlukovania.

14.2 Hierarchické metódy

Hierarchické metódy [2] vytvárajú zhluky na základe hierarchickej dekompozície množiny vstupných údajov. Delia sa na rozdeľovacie a zlučovacie v závislosti od smeru dekompozície. Rozdeľovacie metódy alebo tiež metódy zhora nadol najprv zaradia všetky objekty do jedného zhluku. Potom postupne rozdeľujú tento zhluk na stále menšie zhluky. Naopak, zlučovacie metódy alebo tiež metódy zdola nahor, zaradia každý objekt do vlastného zhluku a potom spájajú zhluky, ktoré sú si najpodobnejšie. Pri obidvoch smeroch sa dekompozícia končí v momente splnenia ukončovacej podmienky.

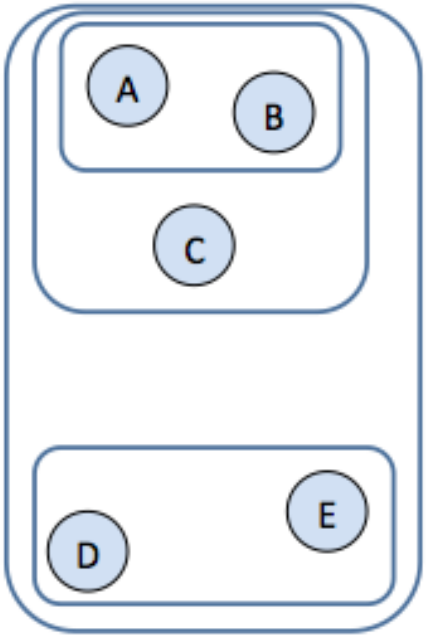
Nevýhodou hierarchických metód je, že ak raz rozdelíme alebo spojíme nejaké zhluky, nemožno tento krok vrátiť späť. Ďalšou nevýhodou je, že táto metóda je výpočtovo veľmi náročná. Preto je niekedy vhodné použiť alternatívne metódy, ako napríklad metóda *K*-priemerov.

Na obrázku 56 vidíme zaradenie podobných položiek do zhlukov použitím hierarchického zhlukovania.

Algoritmus

```
priradiť jeden zhluk každej položke
kým je počet zhlukov > 1
    pre každý zhluk c1
       pre každý zhluk c2 za c1
           vypočítať vzdialenosť medzi c1 & c2
           uložiť tento pár ak majú (doposiaľ) min. vzdialenosť
    spojiť dva najbližšie zhluky
```

Príklad: Hierarchické zhlukovanie



Obrázok 56. Hierarchické zhlukovanie [1].

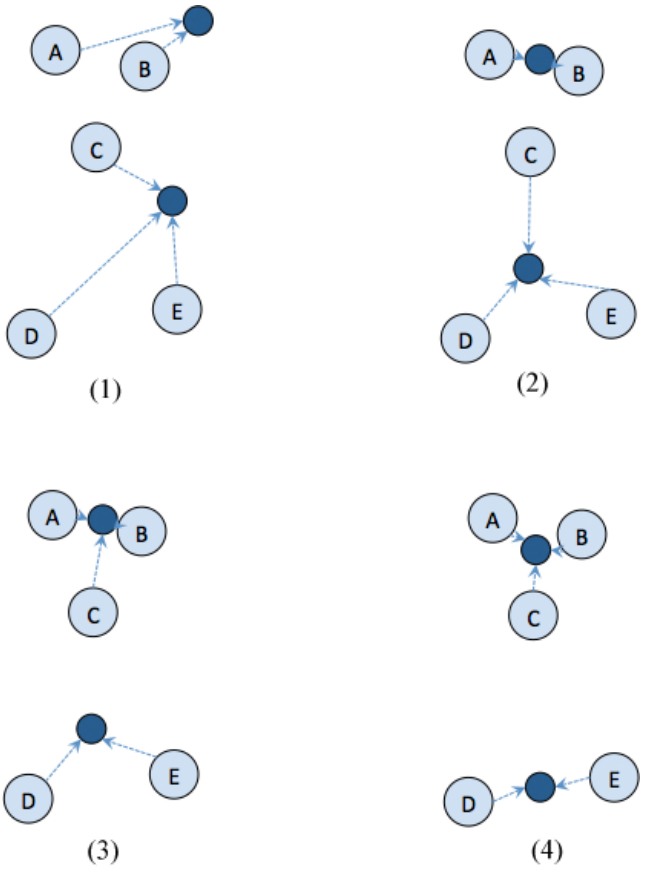
14.3 Metóda *K*-priemerov

Základnou charakteristikou tejto metódy [2] je, že podobnosť jednotlivých objektov a zhlukov sa meria ako ich vzdialenosť vzhľadom na priemernú hodnotu zhluku. Cieľom metódy je minimalizácia zhlukovacieho kritéria, ktorým je najčastejšie súčet štvorcov chýb (rozptylov) všetkých objektov vzhľadom na priemerný objekt. V prvom kroku sa náhodne vyberie *K* objektov, ktoré budú reprezentovať *K* zhlukov. V druhom kroku sa ostatné objekty na základe podobnosti postupne priradia do takto vytvorených zhlukov. Potom sa pre získané zhluky vypočíta nová priemerná hodnota. Druhý krok sa opakuje, pokým dochádza k zmenám v zložení zhlukov. Na obrázku 57 vidíme priebeh tejto metódy.

Algoritmus

```
vložiť K centroidov na náhodné pozície
opakovať kým dochádza k zmenám v priradeniach
    priradiť každú položku k najbližšiemu centroidu
    pohnúť centroid k priemeru priradených položiek
```

Príklad: Metóda K-priemerov



Obrázok 57. Metóda *K*-priemerov [1].

14.4 Vizualizácia zhlukov

Na vizualizáciu zhluknutých údajov poznáme niekoľko metód. V tomto texte spomenieme metódu nazvanú viacrozmerné škálovanie (angl. multidimensional scaling), ktorá slúži na dvojrozmerné zobrazenie viacrozmerných údajov. Táto metóda používa maticu M, ktorej prvky ($M_{i,j}$) obsahujú vzdialenosti medzi i-tou a j-tou položkou. Vizualizované položky sa snažia presúvať takým spôsobom, aby vzdialenosti medzi nimi korešpondovali s hodnotami v matici M.

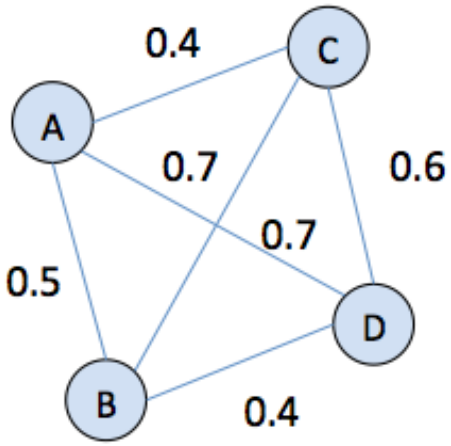
Algoritmus vizualizácie zhlukov

```
rozložiť n položiek na náhodné pozície v dvojrozmernom priestore
opakovať kým chyba* medzi položkami je veľká
   vypočítať vzdialenosti medzi položkami
   posunúť dve položky bližšie alebo ďalej podľa proporcie chyby
```

*chyba – rozdiel medzi hodnotou v matici a pozíciou v priestore

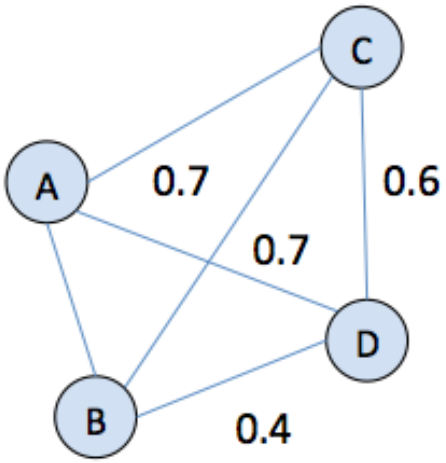
Príklad: Vizualizácia zhlukov

Na obrázku 58 sú náhodne rozložené 4 položky s označenými vzdialenosťami a príslušná matica M s požadovanými vzdialenosťami. Začneme uzlom A. Vidíme, že vzdialenosť medzi uzlom A a uzlom B by mala byť 0,2. Je 0,5 - uzol A priblížime v uzlu B. Následne zistíme, že skutočná vzdialenosť uzlov A a C (0,4) je menšia ako požadovaná (0,8) - uzol C vzdialime od uzlu A. Po týchto krokoch sa dostaneme do stavu, ktorý je zobrazený na obrázku 58. Algoritmus sa opakuje, kým rozdiel medzi hodnotami v matici a skutočnými hodnotami nie je zanedbateľný.



	A	B	C	D
A	0	0.2	0.8	0.7
B	0.2	0	0.9	0.8
C	0.8	0.9	0	0.1
D	0.7	0.8	0.1	0.0

Obrázok 58. Začiatočný stav [1].



Obrázok 59. Stav po prvých dvoch krokoch [1].

14.5 Zhrnutie

Pri práci a analýze veľkého množstva údajov sa objavujú mnohé výzvy. Zhlukovanie údajov pri tejto práci významnou a dôležitou oblasťou. Mnohí výskumníci sa tejto oblasti venujú a snažia sa zlepšovať zhlukovacie algoritmy. V tejto kapitole sme sa venovali niektorým vybraným zhlukovcím metódam a to konkrétne hierarchickým metódam a metóde *K*-priemerov. Na konci kapitoly sme aspoň stručne spomenuli aj vizualizáciu zhlukov. S vďakou uvádzame, že text tejto kapitoly vychádza okrem iných z obsahu prednášky, ktorej autorom je McCrown [1].

Literatúra

[1] McCown, F.: *Introduction to Web Science*. Harding University, (2013). Dostupné na: http://www.harding.edu/fmccown/classes/comp475-s13

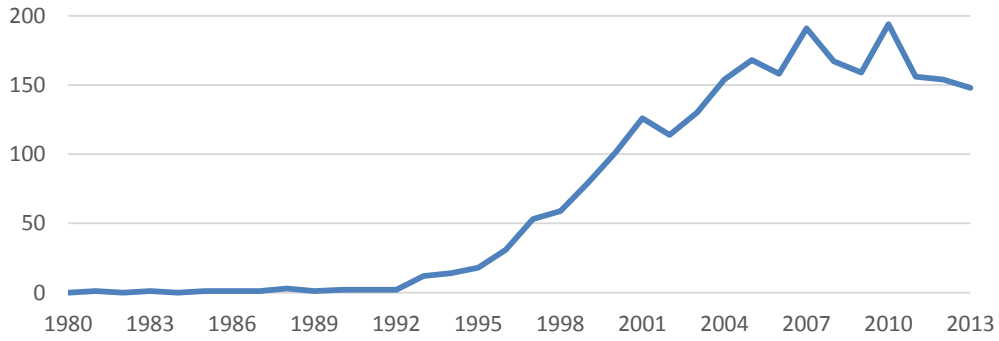
[2] Easly D., Kleinberg J.: *Networks, Crowds and Market: Reasoning About a Highly Connected World*. Cambridge, (2010). ISBN: 9780521195331. Dostupné na: http://www.cs.cornell.edu/home/kleinber/networks-book/

[3] Michálek, M.: *Zhlukovacie algoritmy*. Slovenská technická univerzita, Fakulta informatiky a informačných technológií, (2008). Dostupné na: http://www2.fiit.stuba.sk/~kapustik/ZS/Clanky0809/michalek

15 Filtrovanie dokumentov

V súčasnosti, v dobe preťaženia informáciami, sa používa celá rada postupov na to, aby sme znížili množstvo nepotrebných alebo neužitočných informácií, ktoré súperia o našu pozornosť. Jedným z takýchto postupov je filtrovanie dokumentov, ktorého úlohou je z prúdu dokumentov, ktoré sa na nás hrnú, vybrať len tie, ktoré spĺňajú nejakú vopred stanovenú podmienku. Téma filtrovania dokumentov v ostatnej dobe získala pomerne značné množstvo pozornosti výskumnej komunity, ktorá navrhla množstvo prístupov a aplikácií pre filtrovanie dokumentov. V tejto práci si opíšeme niektoré z týchto prístupov.

Filtrovanie dokumentov je pojem, ktorý svoju najväčšiu popularitu získal po tom, ako sa mu začala venovať konferencia TREC v roku 1995. Od tohto roku postupne stúpal počet publikácií, ktoré sa touto témou zaoberali a až počas ostatných 5 rokov začal počet publikácií kulminovať, čo je viditeľné z grafu počtu publikácií pre termín „document filtering“ získaný pomocou vyhľadávača Google Scholar zobrazení na obrázku 60.



Obrázok 60. Počet publikácií pre dopyt "document filtering" rozdelený podľa rokov publikácie.

Počas tejto doby vzniklo množstvo výskumných prác, ktoré sa zoberali tak prístupmi k filtrovaniu dokumentov ako aj jeho aplikáciami. Rôzni výskumníci navrhli rôzne prístupy k filtrovaniu dokumentov, pričom v tejto práci sa pokúsime vytvoriť ich základný prehľad.

15.1 Čo je to filtrovanie dokumentov

V kontexte filtrovania dokumentov sa často používa viacero súvisiacich termínov, ako je smerovanie dokumentov a vyhľadávanie, ktoré sa častokrát zamieňajú. Je však dôležité ich rozlišovať a poznať rozdiely medzi nimi.

Podľa definície, ktorú vytvoril Lewis v roku 1995 [3] a Hull v roku 1996 [1] je filtrovanie dokumentov proces, na vstupe ktorého je prúd dokumentov a ktorého cieľom je určiť, či jednotlivé dokumenty spĺňajú vopred definovaný dopyt alebo nie.

Smerovanie dokumentov [2] je veľmi podobné ako filtrovanie dokumentov a tieto dva pojmy sa častokrát voľne zamieňajú. Na rozdiel od filtrovania dokumentov pri smerovaní dokumentov sa dá naraz spracovať celá množina dokumentov. Pri filtrovaní dokumentov sa predpokladá, že spracúvaná množina je príliš veľká na to, aby sme ju spracovali celú naraz alebo jednoducho tieto dokumenty pribúdajú postupne, v potenciálne neobmedzenom prúde údajov.

Rozdiel medzi filtrovaním dokumentov a *vyhľadávaním* zas tkvie v tom, že pri filtrovaní máme dopyt stanovený vopred a snažíme sa postupne označovať všetky dokumenty ako vyhovujúce dopytu alebo nevyhovujúce dopytu. Pri vyhľadávaní však dopyt nepoznáme vopred a snažíme sa nájsť dokumenty, ktoré najlepšie vyhovujú stanovenému dopytu až v čase vytvorenia dopytu. Po tom, ako raz získame podmnožinu dokumentov, ktoré vyhovujú dopytu, nás už tento dopyt viac nezaujíma.

15.2 Prístupy k filtrovaniu dokumentov

Pre filtrovanie dokumentov postupne vzniklo množstvo rôznorodých prístupov založených na kategorizácií do dvoch tried (dokument vyhovuje dopytu / nevyhovuje dopytu). Prístupy sa môžu líšiť rôznymi heuristikami, kde je filter definovaný ručne alebo poloautomaticky. Známe je aj kolaboratívne filtrovanie, čo je pomerne špecifická forma filtrovania dokumentov, kde sa dokumenty vyberajú na základe interakcie používateľov s dokumentami a podobnosti medzi používateľmi.

Pri *klasifikácii dokumentov* do dvoch tried je možné použiť rôzne z množstva algoritmov strojového učenia pre klasifikáciu. Využíva sa tam učenie s učiteľom na základe množiny dokumentov, ktoré sú vopred zaradené do zvolených kategórií. Pri riešení tohto problému je možné vybrať niektorý z množstva algoritmov na základe znalosti problému a vlastností algoritmov, ako sú napríklad neurónové siete, SVM, naivný bayesovský klasifikátor alebo rozhodovacie stromy.

Množstvo rôznych aplikácií bolo založených na *ručnom definovaní rôznych filtrov*. Priamu podporu pre takéto filtrovanie zabudovali napríklad do populárneho open-source vyhľadávacieho

nástroja ElasticSearch[40], ktorý poskytuje funkciu perkolátora. Je to ručne definovaná množina dopytov, ktoré sa overujú voči vkladanému dokumentu a v prípade splnenia vyvolávajú notifikáciu. Takáto funkcionalita sa často používa napríklad na detekciu podozrivých dokumentov alebo na automatizované zaraďovanie dokumentov do vopred stanovených kategórií.

Kolaboratívne filtrovanie [6] našlo svoje uplatnenie v rôznych nástrojoch na odporúčanie, kde sa sleduje interakcia používateľov s dokumentami a na základe podobnosti používateľov sa z množiny dokumentov vyberajú také, ktoré by mohli zaujímať daného používateľa. Takýto prístup k odporúčaniu dosahuje pomerne dobré výsledky bez nutnosti znalosti obsahu dokumentu, pričom stačí znalosť o interakcii ostatných používateľov s dokumentom. Tento prístup však do veľkej miery trpí takzvaným problémom studeného štartu vždy, keď do množiny dokumentov pribudne nový dokument, pre ktorý nemáme informácie o interakcii používateľov s ním alebo v prípade, ak do systému pribudne nový používateľ.

Rôzne prístupy k filtrovaniu dokumentov našli svoje aplikácie pri odporúčaní, triedení a kategorizácii dokumentov [7], ohodnocovaní dokumentov [5] ale napríklad aj pri detekcii plagiátov [4].

15.3 Filtrovanie dokumentov ako jednoduchý SPAM filter

V tejto časti si ukážeme príklad, ako sa dá použiť filtrovanie dokumentov ako jednoduchý spam-filter. Detekcia spamu sa dá považovať za problém kategorizácie dokumentov do dvoch tried: je to spam alebo nie je to spam. Pre potreby tohto príkladu budeme používať naivný Bayesov klasifikátor, na ktorom si ukážeme fázu trénovania klasifikátora a fázu klasifikácie dokumentu na jednoduchom príklade.

Na vstupe nášho algoritmu máme trénovaciu množinu dokumentov, o ktorých vieme, či je to spam alebo nie.

```
"We should watch more" → OK
"Do more good to others" → OK
"Poker, blackjack, and casino" → Spam
"Make more money at the online casino" → Spam
"Watch one more time" → OK
```

Vo fáze predspracovania tejto množiny dokumentov vykonáme štandardné kroky, ako je tokenizácia, prevedenie slov do základného tvaru a odstránenie stop-slov. Z viet si vytvoríme množinu vlastností, ktoré sa v našom príklade budú reprezentovať jednotlivými slovami a určíme si, koľkokrát boli jednotlivé vlastnosti v dokumentoch označených ako spam a koľkokrát v užitočných dokumentoch. Získané početnosti sú zhrnuté v tabuľke 3.

[40] ElasticSearch: http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/search-percolate.html

Tabuľka 3. Počet výskytov vlastností v dokumentoch označených ako spam a ako užitočné dokumenty.

	OK	**SPAM**
Casino	0	2
Money	0	1
More	3	1
Watch	2	0

K početnostiam v tabuľke 3 pridáme pravdepodobnosti, že vlastnosť sa nachádza v dokumente označenom ako spam alebo ako užitočný dokument. Táto pravdepodobnosť sa rovná pomeru medzi počtom výskytov slova a počtom dokumentov označených ako spam/užitočný. Pravdepodobnosti pre náš príklad sú zobrazené v tabuľke 4.

Tabuľka 4. Pravdepodobnosti, že vlastnosť sa nachádza v dokumente označenom ako spam/užitočný.

	OK	**SPAM**	**Pr(slovo|OK)**	**Pr(slovo|SPAM)**
Casino	0	2	0/3	2/2
Money	0	1	0/3	1/2
More	3	1	3/3	1/2
Watch	2	0	2/3	0/2

Pre slová, ktoré sa v dokumentoch nachádzajú len veľmi zriedkavo, by mohlo znamenať označenie dokumentu za spam / užitočný, že toto slovo by získalo veľmi vysokú váhu ako indikátor spamu. Preto je bežnou praxou, že sa každému slovu pridelí počiatočná neutrálna váha a tá sa na základe výskytov v dokumentoch upravuje. Na výpočet takejto váhovanej pravdepodobnosti sa používa vzorec:

$Pr_w(\textit{vlastnosť} \mid \textit{trieda})$ = (váha * prvotná pravdepodobnosť + počet dokumentov * Pr) / (počet dokumentov + váha)

kde Pr je pravdepodobnosť, že vlastnosť sa nachádza v dokumente, ktorú sme vypočítali v predchádzajúcom kroku. Váhované pravdepodobnosti pre vlastnosti z nášho príkladu sú zobrazené v tabuľke 5.

Tabuľka 5. Váhované pravdepodobnosti výskytu vlastnosti v triede dokumentov.

	OK	**SPAM**	**Pr(slovo|OK)**	**Pr(slovo|SPAM)**	**Pr_w(OK)**	**Pr_w(SPAM)**
Casino	0	2	0/3	2/2	0.27	0.83
Money	0	1	0/3	1/2	0.25	0.75
More	3	1	3/3	1/2	0.9	0.5
Watch	2	0	2/3	0/2	0.61	0.17

Vo fáze klasifikácie naivný bayesovský klasifikátor kombinuje pravdepodobnosti výskytu jednotlivých vlastností dokumentu v jednotlivých triedach dokumentov a odvodzuje z nich pravdepodobnosť toho, že celý dokument patrí do jednej zo zvolených tried. Na výpočet tejto pravdepodobnosti sa používa Bayesovo pravidlo, ktoré má pre náš príklad túto podobu:

$$Pr(A|B) = \frac{Pr(B|A) * Pr(A)}{Pr(B)}$$

$$Pr(Kat|Dok) = \frac{Pr(Dok|Kat) * Pr(Kat)}{Pr(Dok)}$$

Pri klasifikácii dokumentu „more money“ by výpočet pravdepodobnosti, že dokument je užitočný, vyzeral takto:

```
Pr(OK | "more money")
    = Pr("more money" | OK) * Pr(OK)
    = Pr(more|OK) * Pr(money|OK) * 3/5
    = 0.9 * 0.25 * 0.6
    = 0.135
```

S pravdepodobnosťou 0,135 by sme teda považovali dokument za užitočný. Podobným postupom by sme mohli určiť, s akou pravdepodobnosťou je tento dokument spam. Použitie naivného bayesovského klasifikátora poskytuje niekoľko zaujímavých vlastností pri trénovaní spam-filtra. Patrí medzi ne pomerne vysoká presnosť klasifikácie vzhľadom na jednoduchosť modelu, nutnosť len jediného prechodu cez údaje pri trénovaní modelu alebo jednoduchosť, s akou je možné pridať ďalšie dokumenty do natrénovaného modelu. Tieto vlastnosti robia naivný bayesovský klasifikátor vhodným algoritmom pre túto aplikáciu.

Na pomerne jednoduchom príklade sme si ukázali klasifikáciu dokumentov do dvoch tried pomocou naivného bayesovského klasifikátora, ktorý sa dá pomerne priamočiaro použiť ako jednoduchý spam filter.

15.4 Zhrnutie

V tejto kapitole sme sa venovali problematike filtrovania dokumentov, v ktorej sa snažíme o filtrovanie prichádzajúceho prúdu dokumentov na základe vopred stanovených podmienok. V ostatných rokoch vzniklo pre riešenie tohto problému množstvo rôznych metód založených napríklad neurónové siete, SVM, naivný bayesovský klasifikátor alebo rozhodovacie stromy. Použitie týchto metód sme ilustrovali jednoduchým príkladom naivného bayesovského klasifikátora použitého ako SPAM filter.

Literatúra

[1] Hull, D. A., Pedersen, J. O., Schütze, H.: Method combination for document filtering. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, (1996), pp. 279 - 287.

[2] Hull, D. A., Pedersen, J. O., Schütze, H.: Document routing as statistical classification. In *AAAI Spring Symposium on Machine Learning in Information Access*, (1996).

[3] Lewis, D. D.: Evaluating and optimizing autonomous text classification systems. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, (1995), pp. 246-254.

[4] Meyerzon, D., et al.: Method and system for detecting duplicate documents in web crawls. U.S. Patent No. 6,547,829. 15, (2003).

[5] Persin, M.: Document filtering for fast ranking. In *SIGIR'94*. (1994), pp. 339-348.

[6] Sarwar, B., et al.: Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. (2001), pp. 285-295.

[7] Sebastiani, F.: Machine learning in automated text categorization. In *ACM computing surveys (CSUR)* vol. 34 issue 1, (2002), pp. 1-47.

16 Namiesto záveru

Uzavrieť definitívne otázku, či bude pokus o zavedenie novej vednej disciplíny úspešný, nie je naším cieľom. Lenže práve odpoveď na túto otázku by sa najlepšie hodila do záveru vedeckej publikácie, venovanej webovede. Preto radšej knižku ukončíme bez záveru.

V predchádzajúcich pätnástich kapitolách sme sa pokúsili uviesť čitateľa do problematiky skúmania webu ako javu, ktorý dokázal za menej než štvrťstoročie nadobro zmeniť život väčšiny obyvateľov našej planéty. Najprv sme stručne uviedli východiská, ktoré by mohli viesť k ustanoveniu webovedy ako novej vednej disciplíny. Potom sme sa venovali tomu, čo by mohol byť principiálny predmet skúmania. No a potom sme pokračovali rôznymi metódami, ktoré sa pri skúmaní alebo navrhovaní a prevádzkovaní webu používajú. Vonkoncom si nerobíme nárok, že by sme poskytli ich vyčerpávajúci prehľad. Domnievame sa však, že si čitateľ môže urobiť dobrý obraz o tom, aký výskum webu v súčasnosti prebieha. Oprávňuje nás k tomu snáď tak trochu aj fakt, že v každej z opisovaných tém aj sami aktívne výskumne pôsobíme, aj keď sme úmyselne nekoncipovali túto knižku primárne ako prezentáciu vlastných výsledkov.

Je nesporné, že web bude naďalej treba rozvíjať aj skúmať. Takisto je nesporné, že na jeho komplexné skúmanie treba pohľady viacerých vedných disciplín. Či sa však potreba interdisciplinárneho prístupu pretaví do všeobecne uznaného etablovania novej vednej disciplíny, zostáva nateraz otvorená otázka. Táto knižka je skromným pokusom pomenovať, čo by mohol byť predmet skúmania takej novej vedy – povedzme webovedy, ako sme ju trochu provokatívne nazvali a aké metódy skúmania by jej mohli byť vlastné. Nebolo našou ambíciou dať odpoveď na otázku, či má vzniknúť nová vedná disciplína. Možno tá otázka nie je na celom najpodstatnejšia. Web tu je a bude. Naďalej nás bude fascinovať a klásť pred nás nové výzvy. Vďaka webu žijeme my, výskumníci, ktorí sa ním zaoberáme, v zaujímavých časoch. Nech je táto knižka tak trochu aj pozvaním študovať tento vzrušujúci jav. Pridáte sa?

Index