Identifikace navigační struktury webové prezentace na základě topologie odkazů

Filip Volavka, Vojtěch Svátek

Katedra informačního a znalostního inženýrství, Vysoká škola ekonomická v Praze,
nám. W. Churchilla 4, 130 67 Praha 3
svatek@vse.cz

Abstrakt. Významným prvkem webových prezentací jsou hypertextové odkazy. Vedle ad hoc odkazů se používají pravidelné struktury uspořádané do podoby (často víceúrovňových) nabídek, které označujeme jako navigační struktury. Pozice stránek vzhledem k navigační struktuře je významným indikátorem jejich role v rámci prezentace, a úplnost navigační struktury napovídá o kvalitě celého návrhu. Proto má smysl se navigační strukturou zabývat v rámci komplexní analýzy webových prezentací. Článek popisuje metodu identifikace navigační struktury založenou čistě na topologii odkazů, a obsahuje výsledky jejího testování na reálných datech.

Klíčová slova: WWW, topologie

1 Úvod

Topologie hypertextových prezentací je předmětem formální analýzy již po několik desetiletí. Standardně používané techniky vycházejí z teorie grafů: jednotlivé stránky jsou chápány jako uzly a odkazy mezi nimi jako hrany v orientovaném grafu. Na tuto reprezentaci lze pak aplikovat metriky, mezi které patří např. kompaktnost nebo stratum dané prezentace (stručný přehled lze nalézt např. v [10]).

Nový pohled na topologii hypertextu otevřel v 90. letech vznik WWW s jeho koncepcí neohraničeného hypertextového dokumentu. Tradiční hypertextové prezentace byly zpravidla vytvořeny jedním autorem (ev. kolektivem), obsahovaly pevně daný počet stránek, věnovaly se jednomu tématu, a kladly důraz na integritu odkazů. Charakteristickými rysy webu bylo naopak od počátku nezávislé autorství mnoha osob, neustálá proměnlivost, propojenost (téměř) všeho se vším, a nestabilita odkazů. Přístupy k analýze tzv. *webgrafu* proto začaly klást důraz na problém objevení souvislostí mezi nezávisle vzniklými dokumenty v otevřeném prostoru. Zřejmě nejvýznamnějšími výsledky tohoto výzkumu se staly algoritmy pro zjišťování popularity stránek (PageRank, HITS, viz např. [5]) a poznatky o globální struktuře webu; rozšířenou úlohou je rovněž hledání „odborných komunit“. Problematice webgrafu je věnována série specializovaných workshopů, v r. 2003 např. při World–Wide Web Conference [1].

Důraz kladený “webgrafovou” komunitou na techniky schopné efektivně analyzovat topologické struktury v prostoru celého WWW je vcelku pochopitelný.

Přesto se zdá, že problematika lokálních prezentací zůstává poněkud neprávem opomíjena. Z jednotlivých (zejména firemních) prezentací[1] lze totiž extrahovat informace použitelné např. pro rozhodování o nákupech výrobků či volbě obchodního partnera. Pro získání těchto informací je nutné analyzovat plný text stránek, informace o jejich topologické poloze v rámci prezentace však může analýzu usnadnit a zefektivnit. Přitom jsou data potřebná pro topologickou analýzu k dispozici bez zvláštního úsilí – seznamy odkazů jsou totiž vedlejším produktem činnosti webového „robota“, který stránky z webu stahuje.

Významným aspektem, kterým se analýza webových prezentací do jisté míry liší jak od globálních “webgrafových” přístupů, tak i od analýzy tradičního hypertextu, je role (často víceúrovňových) *menu*. Webové prezentace jsou budovány relativně centralizovaně a zejména ve firemním prostředí bývají podřizovány jednotnému menu. Toto menu bývá současně jediným nástrojem navigace, na rozdíl od specializovaných hypertextových systémů, které pro tento účel nabízejí různé nadstavby. Pro naši potřebu si proto zavedeme pojem *navigační struktury* webové prezentace; ta je složena ze samostatných *komponent* odpovídajících dílčím menu. Každá stránka, na kterou je z menu odkazováno, v sobě celé toto menu současně obsahuje, zpravidla v zobrazení evokujícím “lištu”. Navigační struktura tedy uživateli umožňuje “horizontální” pohyb mezi stránkami ležícími na stejné úrovni hierarchie – z topologického hlediska je proto komponenta navigační struktury množinou stránek vzájemně propojených systémem každá s každou.

Přestože navigační struktura v uvedeném pojetí není jedinou možností, jak přístup k webové prezentaci organizovat, je alespoň v případě firemních webů jednoznačně nejčastější (podle výsledků uvedených v tomto článku i podle nezávisle provedené studie [11] se jedná o zhruba 60%), a v současnosti výrazně převládá např. nad prostými hierarchiemi (bez “horizontálního” propojení) nebo využíváním rámců. Její *automatická identifikace* je proto úlohou s velmi frekventovaným využitím. Zatímco v plném textu (resp. kódu HTML stránky) se ovšem nabídkové lišty rozpoznávají poměrně obtížně, v topologii odkazů by jejich struktura měla být snadno patrná. V předkládaném článku se pokoušíme tuto hypotézu ověřit pomocí experimentů na reálných webových datech. Podrobnější informace o projektu lze nalézt v [14].

Článek má následující strukturu. V kap. 2 vysvětlujeme použitý algoritmus a odhadujeme jeho složitost. V kap. 3 popisujeme způsoby implementace algoritmu. V kap. 4 uvádíme výsledky dosažené na datech odvozených z prezentací firem “náhodně” vybraných z rozsáhlé množiny odkazované webovým katalogem. V kap. 5 rozebíráme perspektivy praktické využitelnosti metody. V kap. 6 porovnáváme náš přístup s několika obdobnými projekty. Závěrečná kap. 7 pak obsahuje shrnutí přínosů i slabin metody.

[1] Běžně se používá i přibližně ekvivalentní pojem *webové sídlo* (“website”). Termín *prezentace* však podle našeho názoru lépe vystihuje skutečnost, že jejím předmětem nemusí nutně být jednotlivá firma, ale např. i výrobek nebo služba.

2 Algoritmus pro identifikaci navigačních struktur

Algoritmus identifikující navigační strukturu v pojetí vymezeném v úvodní kapitole zahrnuje několik dílčích úloh:

1. Získání množiny stránek tvořící webovou prezentaci
2. Určení kořenové stránky prezentace
3. Určení hierarchické úrovně jednotlivých stránek
4. Nalezení komponent navigační struktury pro jednotlivé úrovně.

Úloha č.1 může být za jistých okolností netriviální, např. pokud je firemní prezentace umístěna v souborovém adresáři jiné organizace. Tento problém jsme si v naší práci zjednodušili požadavkem, že výchozí URL pro analýzu nesmí obsahovat názvy vnořených adresářů. Pro úlohu č. 2 v obecném chápání nabízí prostředky opět teorie grafů. První experimenty však ukázaly, že spolehlivost hledání kořenové stránky ve webové topologii není příliš vysoká. Navíc je kořenová stránka obvykle na začátku analýzy známa – v našich experimentech jsme pracovali s množinou prezentací odkazovaných ve webovém katalogu právě prostřednictvím kořenových stránek. Ani touto úlohou se zde proto nebudeme podrobněji zabývat, a omezíme se jen na popis určení hierarchické úrovně stránek a nalezení komponent navigační struktury.

2.1 Podúloha určení hierarchické úrovně stránky

Tato úloha se skládá ze tří částí:

Výpočet matice sousednosti. Matici o rozměrech $n \times n$ (kde n je počet stránek v prezentaci) lze snadno sestavit na základě seznamu odkazů. Prvek x_{ij} matice je roven jedné tehdy, když existuje hrana z uzlu i do uzlu j (tedy stránka i odkazuje na stránku j), jinak je roven nule.

Výpočet matice nejkratších vzdáleností. Prvek x_{ij} v matici nejkratších vzdáleností udává vzdálenost stránky j od stránky i. Tento prvek tedy udává nejmenší počet odkazů, které je nutno projít, aby se návštěvník dostal ze stránky i na stránku j. Pokud stránka j není dosažitelná ze stránky i, potom má prvek x_{ij} hodnotu nekonečno. Postup výpočtu matice nejkratších vzdáleností Floyd–Warshallovým algoritmem (běžně používaným v teorii grafů) jsme převzali z [13]. Jedná se o iterativní výpočet; v q-tém kroku (pro q začínající od 1) se pro všechny uzly i a j počítá matice nejkratších cest z i do j o délce maximálně 2^q, jako *minimální součet vzdáleností* z i do k a z k do j přes všechny uzly k (různé od i,j):

$$d_{ij}(q) = \min_k \left(d_{ik}(q-1) + d_{kj}(q-1) \right)$$

Vzhledem k exponenciálnímu nárůstu délky zkoumaných cest algoritmus skončí nejpozději po přibližně $\log_2 n$ krocích iterace (často ovšem dříve, pokud se matice v daném kroku už nezmění).

Výpočet hloubky jednotlivých stránek. Hloubka stránek v hypertextu se obvykle chápe jako *nejkratší* vzdálenost od kořene [4]. Tohoto chápání jsme se přidrželi i my.

Tabulka 1. Příklad matice sousednosti, matice nejkratších vzdáleností a konvertované matice nejkratších vzdáleností.

	1	2	3	4	5	6
1	0	1	1	1	1	0
2	0	0	1	1	1	1
3	0	1	1	1	1	1
4	0	1	1	0	1	1
5	0	1	1	1	0	1
6	0	0	0	0	0	0

1	2	3	4	5	6
0	1	1	1	1	2
inf	0	1	1	1	1
inf	1	0	1	1	1
inf	1	1	0	1	1
inf	1	1	1	0	1
inf	inf	inf	inf	inf	0

1	2	3	4	5	6
0	1	1	1	1	2
6	0	1	1	1	1
6	1	0	1	1	1
6	1	1	0	1	1
6	1	1	1	0	1
6	6	6	6	6	0

Ve webové prezentaci se sice může stát, že odkaz nepatřící k vlastní navigační struktuře (např. upozornění na „novinky“) vede na stránku, která by jinak byla dosažitelná jen prostřednictvím několika odkazů, provedené experimenty však naznačují, že tento případ nebývá běžný[2]. Z matice nejkratších vzdáleností použijeme jen informaci vztahující se ke kořenové stránce[3].

Vedlejším produktem uvedených výpočtů je *míra kompaktnosti* webové prezentace. Ta není potřebná při vyhledávání navigační struktury, ale poskytuje dodatečnou informaci zejména pro hodnocení kvality návrhu prezentace – příliš nízké hodnoty způsobují zdlouhavou navigaci, příliš vysoké hodnoty mohou vést k zahlcení uživatele množstvím odkazů. Počítá se (podle [10]) z *konvertované* matice nejkratších vzdáleností, ve které jsou nekonečné vzdálenosti nahrazeny konverzní konstantou, standardně celkovým počtem uzlů. Kompaktnost je vlastně normalizovaným rozdílem skutečného součtu hodnot v této matici (*Sum*) a teoretického maxima (*Max*) tohoto součtu pro dané *n*:

$$\mathrm{Cp} = \frac{\text{Max-Sum}}{\text{Max-Min}}$$

V Tab. 1 uvádíme příklad matice sousednosti, matice nejkratších vzdáleností a konvertované matice nejkratších vzdáleností pro webovou prezentaci[4] o šesti stránkách. Pokud budeme jako kořenový uzel grafu chápat stránku č. 1, budou stránky 2 až 5 mít hloubku 1, a stránka 6 bude mít hloubku 2 (tato informace je přímo obsažena v prvním řádku druhé matice). Míra kompaktnosti je

$$\mathrm{Cp} = \frac{180\text{-}76}{180\text{-}30} \cong 0{,}69$$

---

[2] Na často aktualizované stránky prezentací obvykle směřuje trvalý odkaz z hlavní stránky; dodatečný odkaz má pak spíše motivovat k jejich navštívení, než je hypertextově přiblížit.

[3] Předchozí výpočet matice nejkratších vzdáleností by se proto dal zjednodušit. Výhodou použitého řešení je však možnost přímo aplikovat např. algoritmus hledání kořenové stránky, pokud bychom tuto informaci neměli k dispozici. Matici nejkratších vzdáleností chápeme jako výchozí bod analýzy webové topologie, a proto prozatím její výpočet implicitně zařazujeme.

[4] Data jsou převzata ze skutečné prezentace na adrese `http://www.cmiinsulation.com/`.

2.2 Podúloha nalezení komponent navigační struktury

Pro každou množinu stránek, které mají stejnou hloubku, se hledají její *maximální podmnožiny C* o kardinalitě vyšší než 2, a to takové, že pro každou dvojici stránek p_i, $p_j \in C$ existuje odkaz z p_i do p_j i naopak. Takovou C pak označíme za *komponentu navigační struktury*. Jako *navigační strukturu* pak chápeme sjednocení všech komponent navigační struktury obsažených v prezentaci. Toto chápání je poněkud zjednodušující, protože za jistých okolností může fyzická prezentace obsahovat několik samostatných (oddělených) struktur menu; tato skutečnost by se mohla rozpoznat podle výskytu komponent v nesouvislé posloupnosti hloubek. Takové případy je však nepochybně velmi vzácné.

V uvedeném příkladu je na první pohled patrná navigační struktura o jediné komponentě, zahrnující stránky č.2 až 5.

2.3 Odhad algoritmické složitosti

V této části se pokusíme odhadnout časovou a prostorovou složitost s celým výpočtem spojenou. Odhad je spíše neformální, a vychází z programového kódu, který není vždy optimalizován na rychlost. Časově nejnáročnější je výpočet matice nejkratších vzdáleností Floyd–Warshallovým algoritmem, který v nejhorším případě iteruje $log_2 n$-krát přes výpočet d_{ij} prováděný pro všechna i,j,k z celkového počtu n uzlů. Pesimistický odhad jeho časové složitosti je proto

$$C(n) \approx O(n^3 \log_2 n)$$

Celkový odhad časové složitosti navíc zahrnuje čtyři průchody matic o velikosti $n \times n$: plnění matice sousednosti, její převod do tvaru vhodného pro výpočet nejkratších vzdáleností, plnění matice cest nejkratších vzdáleností, a průchod maticí sousednosti při hledání kandidátů na komponenty navigační struktury. Dále jsou prováděny dva průchody vektoru délky n (seznamu všech stránek) při jejich seřazení podle hloubky. Velmi hrubý výsledný odhad časové složitosti je proto

$$C(n) \approx O(n^3 \log_2 n + 4n^2 + 2n)$$

Prostorová složitost je $P(n) \approx O(4n^2)$ vzhledem ke skutečnosti, že pracujeme celkem se čtyřmi maticemi o velikosti $n \times n$.

3 Implementace algoritmu

Vzhledem k různorodým požadavkům byl algoritmus implementován ve dvou variantách: jako program s rozhraním HTML a jako webová služba. Jádrem byl v obou případech stejný program vytvořený ve skriptovacím jazyce PHP, zvoleném právě pro možnost snadného vytvoření rozhraní v HTML. Program s rozhraním HTML slouží pro demonstrační účely, zatímco webová služba je chápána jako prototyp nástroje, který bude v dohledné době začleněn do architektury systému *Rainbow* [12], určeného pro vícecestnou analýzu obsahu a struktury webu.

Tabulka 2. Základní rozdělení zpracovávaných prezentací.

Typ	Počet	Podíl
Prezentace o jedné stránce	15	36 %
Prezentace o více stránkách – úspěšná analýza	24	57 %
Prezentace o více stránkách – neúspěšná analýza	3	7 %

Demonstrační program *s rozhraním HTML* se aktivuje pomocí libovolného webového prohlížeče na adrese `http://rainbow.vse.cz/topo/test.php`. Pro jednoduchost nabízí pouze výběr ze 42 prezentací, uložených v databázi spravované modulem zdrojových dat [6] systému *Rainbow*. Po získání dat prostřednictvím protokolu SOAP [3] a provedení analýzy postupně vypíše:

- seznam všech nalezených komponent navigační struktury, včetně jejich hloubky
- matici sousednosti, včetně její transponované varianty
- matici nejkratších vzdáleností a konvertovanou matici nejkratších vzdáleností
- míru kompaktnosti
- matici cest nejkratších vzdáleností
- (v případě menšího počtu stránek) grafickou reprezentaci[5] webu včetně barevného odlišení vstupní stránky a navigační struktury, viz Obr. 1.

Zatímco demonstrační program využívá pouze klientskou část protokolu SOAP[6], *webová služba* zahrnuje i serverovou část, jejímž prostřednictvím poskytuje výsledná data. Základním výstupem je seznam nalezených stránek patřících do téže komponenty navigační struktury, jako stránka zadaná na vstupu[7]. Výsledky v tomto případě neobsahují průběžně počítané matice a samozřejmě grafickou reprezentaci.

4 Experimentální výsledky

4.1 Vstupní data

Pro testování algoritmu byla použita data ze 42 webových prezentací odkazovaných z větve „Business“ veřejného katalogu *Open Directory*[8]. Kritéria výběru byla dvě:

- Nabídka firmy se týká *produktů materiální povahy*[9].
- Vstupní stránkou prezentace je *kořenové URL*, nejedná se tedy o např. prezentaci viditelně umístěnou na serveru jiné organizace.

Prezentace splňující tato kritéria již byly vybírány „náhodným“ způsobem, bez dalšího filtrování (např. podle typu nabízené komodity, velikosti prezentace nebo technologie použité pro její tvorbu). Data nebyla určena pouze pro analýzu topologie,

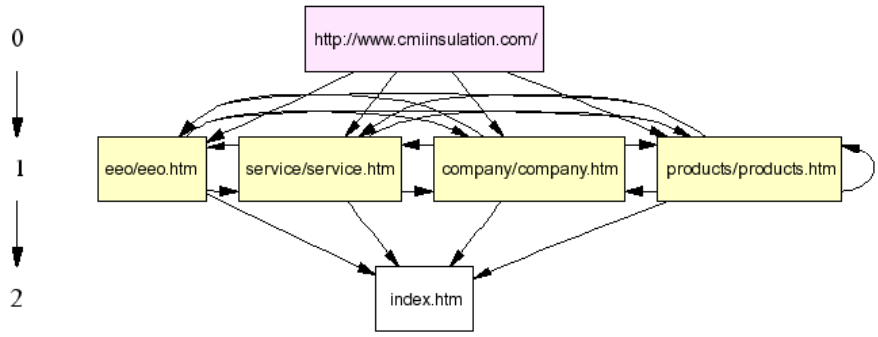
[5] Pro výpis grafické reprezentace je používán volně šířený grafický balík *Graph_Viz*, dostupný na adrese `http://www.research.att.com/sw/tools/graphviz/`.
[6] Implementováno pomocí knihovny SOAP z databáze PEAR, `http://pear.php.net/`.
[7] Jako kořenová stránka je v tomto případě předpokládána stránka přístupná pomocí URL s odstraněnou adresářovou i souborovou částí. Tato část programu by se snadno dala vylepšit o heuristiky identifikující kořenovou stránku pomocí názvu a přípony souboru.
[8] `http://www.opendir.org`
[9] Pracovně je proto označujeme zkratkou OOP – “organisations offering products”.

ale také pro další metody vyvíjené v rámci projektu *Rainbow*; ostatní metody ovšem vesměs pracují s plnými texty stránek, které rovněž zpřístupňuje zmíněný modul správy zdrojových dat. Poněkud překvapivě se 15 z celkového počtu 42 prezentací skládalo jen z jediné stránky, analýzu topologie proto nebylo možné aplikovat. U tří ze zbývajících 27 případů naopak skončil běh programu neúspěšně vzhledem k velkém rozsahu prezentace (problém souvisel s omezením dané implementace na 1000 odkazů zpracovávaných v matici, a je tudíž v principu řešitelný). Shrnutí je uvedeno v Tab. 2.

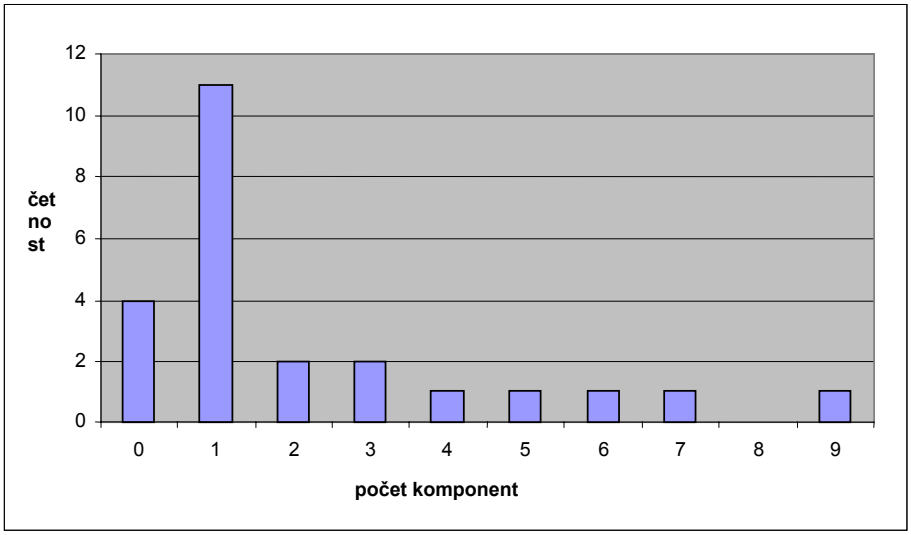
4.2 Vlastní výsledky analýzy

Výsledky analýzy topologie jsme tedy získali pro 24 prezentací. Jejich detailní přehled včetně URL jednotlivých prezentací je uveden v příloze, v uspořádání podle celkového počtu stránek sestupně. Zde se zastavíme u některých souhrnných faktů.

Průměrný počet stránek na prezentaci byl 52. Pouze u 4 prezentací se nepodařilo identifikovat navigační strukturu. *Maximální hloubka* navigační struktury byla v průměru přibližně 3. Maximální dosažený *počet komponent* byl 9, průměrně vycházely asi 2 komponenty na prezentaci. Distribuce četnosti počtu komponent je uvedena v grafu na Obr. 2. Jedna komponenta v průměru obsahovala přibližně 6 stránek, což odpovídá představě "rozumně velkého" menu. *Míra kompaktnosti* byla u našeho vzorku vesměs vyšší než hodnota 0,5 doporučovaná podle [10] pro běžný hypertext (v průměru 0,67). Skutečně by se dalo předpokládat, že tvůrci WWW budou preferovat rychlou dostupnost informací na úkor pravidelné strukturovanosti prezentací. Možná významnějším důvodem je skutečnost, že v klasickém hypertextu se do výpočtu kompaktnosti nezahrnuje rejstřík a referenční uzly (jako explicitní navigační struktura) – na webu jsou ovšem stránky navigační struktury často totožné s obsahovými stránkami, takže je nelze při výpočtu oddělit. Na pohled se ani nezdá, že by hodnota míry kompaktnosti výrazně korelovala s celkovým počtem uzlů prezentace nebo s velikostí navigační struktury v rámci prezentace. Některé další postřehy a podrobněji zpracované údaje k této otázce jsou uvedeny v [14].



Obr. 1. Příklad grafického znázornění analyzované prezentace



Obr. 2. Distribuce četnosti počtu komponent navigační struktury

Příčinou nenalezení navigační struktury (ve 4 případech) bylo:

- U prezentace č.3 použití rámců. Je nutno poznamenat, že jejich výskyt v pouze jediné prezentaci je poměrně překvapivým výsledkem.
- U prezentace č. 23 skutečnost, že v době shromažďování dat pro naši analýzu byla prezentace ve stavu rozpracovanosti. Jak jsme ověřili, později byla navigační struktura do prezentace zařazena.
- Zbývající dvě prezentace, č. 2 a č. 24, skutečně navigační strukturu v žádném smyslu slova neobsahují; jejich strukturu lze označit jako "degenerovanou".

Analyzované prezentace jsme následně podrobili *vizuální kontrole*. Všechny komponenty navigačních struktur nalezené algoritmem ve webovém prohlížeči odpovídaly strukturám interpretovatelným jako menu. Častým problémem analýzy, na který jsme při vizuální kontrole narazili, byla však *synonymie URL*. K ní dochází u vstupních stránek s názvem např. `index.html`, které jsou nabízeny jako implicitní při zadání URL bez uvedení souboru. Z hlediska analýzy topologie jsou ovšem obě URL (s uvedením souboru a bez něj) odlišná, a bude s nimi pracovat jako se dvěma různými stránkami. V příkladu uvedeném na Obr. 1 se tak jediná fyzická vstupní stránka rozdělila do dvou uzlů v hloubce 0 resp. 2. Synonymii URL plánujeme v budoucnu potlačit pomocí ukládání kontrolního "součtu" stránky vypočteného pomocí tzv. *Message Digest* (MD5) algoritmu. Návrh je popsán v [6], avšak nebyl dosud plně realizován.

5 Perspektivy využití

Technika identifikace navigační struktury webové prezentace je poměrně obecná a lze předpokládat její využití pro celou řadu odlišných nadstavbových úloh.

Původní motivací zařazení topologické analýzy v rámci rozsáhlejšího projektu analýzy obsahu a struktury WWW [12] byl její příspěvek k *sémantické klasifikaci* stránek, popřípadě i odkazů. Metoda integrace ontologií navržená v [7] by mohla být

nástrojem, jak čistě topologické kategorie (např. „stránka s mnoha externími odkazy“) asociovat s kategoriemi obsahovými (např. „stránka referencí na zákazníky“). Některé topologické kategorie mohou vycházet právě z příslušnosti resp. blízkosti dané stránky k určité komponentě navigační struktury.

Další aplikační úlohou, kterou jsme se již snažili s identifikací navigační struktury propojit, je odlišení různých tzv. *logických dokumentů*, zejména pak vymezení “jádra” firemní prezentace vůči speciálním částem (např. připojené online knihy, vstupní stránky privátních částí webového sídla). Jednoduchou heuristikou byl v tomto případě předpoklad propojení každé ze stránek jádra na všechny stránky alespoň jedné komponenty navigační struktury. První testy na šesti malých prezentacích naznačily slibné výsledky [15], na datech OOP (obsahujících i rozsáhlejší prezentace) však bylo kvůli chybějícímu odkazu na některou ze stránek komponenty vyřazeno příliš mnoho stránek, které do jádra prezentace z věcného hlediska patřily. Není ovšem vyloučeno, že sofistikovanější metoda rozpoznání jádra by přinesla výsledky lepší.

Poslední úlohou, kterou zmíníme, je hodnocení *kvality návrhu* prezentace. Pro ni lze komplementárně použít na jedné straně uniformní míru kompaktnosti, která je vedlejším produktem výpočtu, jednak heuristiky založené právě na existenci navigační struktury a na jejích vlastnostech – velikosti vzhledem k rozsahu celé prezentace, počtu stránek připadajících na komponentu apod.

6 Srovnání s jinými projekty

Attardi [2] využívá topologickou analýzu webové prezentace jako podpůrný nástroj obsahové klasifikace stránek; cílem je odstranění částí prezentace, které nenesou užitečnou informaci, např. úvodní stránky, nápovědy, stránek pro vyhledávání nebo s reklamou. Všechny odkazy, které se vyskytují na nejméně 90% stránek z celkového počtu analyzovaných stránek, považuje za odkazy tvořící strukturu (“structural links”) a stránky takto odkazované vyřazuje z procesu klasifikace. Nepracuje tedy explicitně se strukturou prezentace. Tento postup je časově nenáročný, nelze ho ovšem aplikovat na velkou část firemních prezentací – ty totiž často obsahují důležité informace i na stránkách patřících do navigační struktury. U malých prezentací může pak snadno dojít k vyřazení všech stránek.

Mathieu a Viennot [8] popisují strukturu webu dvojitě – pomocí odkazů, i pomocí souborové hierarchie odvozené z analýzy URL. Vycházejí z předpokladu, že autoři prezentací se snaží organizovat své soubory přehledně a tato organizace je úzce svázána s logickou strukturou webu. Ve svém experimentu seřadili 8 milionů URL z domény .fr podle cesty k souboru, a vypočítali matici sousednosti. Prezentace pak chápou jako shluky podél diagonály matice sousednosti, takové, že většina odkazů ze stránek shluku vede opět do téhož shluku. Opět se jedná o rychlou metodu, což je podmínkou jeho použití pro takto rozsáhlý vzorek. Jejím problémem je ovšem závislost na předpokladu dodržování pravidel pro umístění souborů do adresářů s ohledem na pozici v topologii odkazů. Při aktualizaci prezentací se tato pravidla v praxi často nedodržují, navíc se v současnosti začínají pro předávání informací používat “virtuální URL”, která fyzické struktuře adresářů vůbec neodpovídají.

Metoda je navíc opět závislá na kvantitativní heuristice vyjadřující poměr odkazů dovnitř a vně shluku.

Vedle konkrétních výzkumných projektů se zmíníme i o obecné grafové technice hledání *silně souvislých komponent* (SCC), popsané např. v [9]. SCC jsou podgrafy daného grafu takové, že pro každé dva uzly v SCC existuje *cesta* z jednoho do druhého a zpět; algoritmicky jsou zjišťovány z matice nejkratších vzdáleností. Obecnou techniku hledání SCC na webové prezentace nelze použít, protože všechny stránky bývají navzájem dosažitelné, a celá prezentace je tudíž jedinou SCC. Náš přístup proto používá tuto matici pouze pro zjišťování hloubky uzlů, zatímco vlastní tvorba komponent je založena na *přímých odkazech* uvedených v matici sousednosti.

7 Závěr

Navržená technika identifikace navigační struktury se v experimentech ukázala jako použitelná na podstatnou část firemních webových prezentací. Může se proto stát východiskem pro sofistikovanější postupy, směřující zejména k odvození sémantické informace a tím i k podpoře např. textově orientovaných metod analýzy WWW.

Slabinou současné implementace metody je relativně pomalá odezva, zejména pro rozsáhlejší webové prezentace – pohybuje se řádově v desítkách sekund až jednotkách minut. Její příčinou je, při poměrně přijatelné složitosti algoritmu (viz kap. 2.3), zejména nutnost získávat při každém volání data prostřednictvím protokolu SOAP, nižší efektivita interpretovaného jazyka PHP oproti kompilovaným jazykům, a také počítání kompletního „učebnicového“ souboru výsledků (vč. např. míry kompaktnosti, konvertované matice atd.), který v praxi často nemusí být potřebný. Předpokládáme, že v další verzi modulu pro analýzu topologie, již plně integrované do architektury Rainbow, budou poslední dva aspekty zohledněny.

Autoři by rádi poděkovali M. Sajalovi za přínos k řešení projektu, a V. Snášelovi za podnětné připomínky v závěrečné fázi práce. Poděkování též patří anonymním recenzentům článku. Projekt je částečně podporován grantem GAČR 201/03/1318 „Inteligentní analýza obsahu a struktury WWW“.

Reference

1. *Second Workshop on Algorithms and Models for the Web-Graph* (WAW 2003), `http://www.almaden.ibm.com/cs/people/ravi/waw2003.html`.
2. Attardi G., Gull A., Sebastiani F. Automatic Web Page Categorization by Link and Context Analysis. In: *THAI-99, European Symposium on Telematics, Hypermedia and Artificial Inteligence*, ed.: Chris Hutchison and Gaetano Lanzarone, Varese, 1999, s. 105–119.
3. Box D. et al. *Simple Object Access Protocol (SOAP)* 1.1 -- W3C Note. W3C, 2000. `http://www.w3.org/TR/SOAP`

4. Botafogo R., Shneiderman B. Identifying Aggregates in Hypertext Structures, *UK Conference on Hypertext*, 1991.
5. Chang G., Healy M.J., McHugh J.A.M., Wang J.T.L. *Mining the World Wide Web*. Kluwer Academic Publisher 2001. ISBN 0-7923-73-49-9
6. Kosek J. *Inteligentní podpora navigace na WWW s využitím XML*. [Diplomová práce], Vysoká škola ekonomická v Praze, 2002. Dostupné na WWW: `http://www.kosek.cz/diplomka`.
7. Labský M., Svátek V. Ontology Merging in Context of Web Analysis. In: *Workshop DATESO'03*, VŠB–TU Ostrava 2003.
8. Mathieu F., Viennot L.: Local Structure in the Web. In: *Poster Session of the International World–Wide Web Conference*, Budapest 2003.
9. Plesnik, J.: *Grafové algoritmy*. Veda, Bratislava 1983.
10. Pokorný J., Snášel V., Húsek D. *Dokumentografické informační systémy*. 1.vyd., Praha, Karolinum 1998.
11. Sajal M. *Analýza topologie odkazů mezi webovými stránkami*. [Diplomová práce], Vysoká škola ekonomická v Praze, 2002.
12. Svátek V., Kosek J., Labský M., Bráza J., Kavalec M., Vacura M., Vávra V., Snášel V. Rainbow – Multiway Semantic Analysis of Websites. In: *The 2nd DEXA International Workshop on Web Semantics,* Praha 2003, IEEE Computer Science Press, 2003, s. 635–639. ISBN 0-7695-1993-8.
13. Vejmola S. *Teorie grafů*. 1. vyd. Praha, VŠE v Praze 1985.
14. Volavka F. *Identifikace logického jádra webového sídla na základě topologie odkazů* [Diplomová práce], Vysoká škola ekonomická v Praze, 2003.
15. Volavka F., Sajal M., Svátek V. Topology–based discovery of navigation structure within websites. In: Popelínský L. (ed.) *DATAKON 2003.* Brno, Masarykova univerzita, 2003, s. 295–300. ISBN 80-210-3215-4.

Annotation:

Identification of website navigation structure based on link topology

Beside ad hoc hyperlinks, websites typically contain regular navigation structures corresponding to (often hierarchical) menu bars. Position of pages with respect to navigation structure partially indicates their role within the site, and the completeness of the structure reflects the quality of overall design. We describe a simple method for navigation structure discovery purely based on link topology, and present its test results on real–world data.

Příloha: seznam výsledků na datech OOP s uvedením adres URL

	Startovní adresa	Počet stránek	Velkost NS	Počet komponent NS	Rozmezí hloubek NS	Prům. velikost komp.	Míra kompaktnosti	Max hloubka NS
1	www.adcom.com	170	32	5	1-3	6,4	0.26	4
2	www.woodencanoe.com	139	43	9	2-4	4,8	0.96	5
3	www.idealindustries.com	137	0	0	0		0.09	4
4	www.transducertechniques.com	91	18	3	1-2	6	0.91	4
5	www.harris-bruno.com	81	17	4	2-3	4,3	0.74	4
6	www.pipingspecialties.com	75	4	1	1	4	0.05	3
7	www.blecha.com	66	5	1	1	5	0.91	3
8	www.aseapower.com	66	10	1	1	10	0.94	3
9	www.indcommutator.com	60	46	6	1-2	7,7	0.83	3
10	www.precisionstampedconcrete.com	52	7	1	1	7	0.93	2
11	www.ewbank.co.uk	48	16	2	2-3	8	0.92	4
12	www.ahiroofing.com	44	12	1	1	12	0.65	3
13	www.goodmansinc.co	40	18	3	1-2	6	0.62	3
14	www.radesrl.com	40	36	7	1-2	5,1	0.95	2
15	www.kanegrade.com	30	4	1	1	4	0.92	3
16	www.nrdstaticcontrol.com	22	9	1	1	9	0.84	3
17	www.talbrospolymers.com	20	8	1	1	8	0.68	6
18	www.prolining.com	17	9	2	1-3	4,5	0.83	3
19	www.hawleys.com.au	15	9	1	1	9	0.83	5
20	www.lonearrow.com	13	3	1	1	3	0.27	2
21	www.cmiinsulation.com	6	4	1	1	4	0.69	2
22	www.brick1.com	6	0	0	0		0.30	1
23	www.tongba.com	3	0	0	0		0.33	1
24	www.fwftrucksales.com	2	0	0	0		0.50	1