Určenie podobnosti inštancií ontologických konceptov pre adaptívne aplikácie založené na webe so sémantikou

Anton Andrejko, Mária Bieliková

Ústav informatiky a softvérového inžinierstva
Fakulta informatiky a informačných technológií
Slovenská technická univerzita, Ilkovičova 3, 842 16 Bratislava
{andrejko,bielik}@fiit.stuba.sk

Abstrakt Ak používateľovi prezentujeme dva dokumenty a on ohodnotí ich obsah explicitne (napr. záujem o ne), porovnaním a ďalším skúmaním spoločných a odlišných atribútov týchto dokumentov, môžeme získať zaujímavé informácie o záujmoch používateľa. V aplikáciách pre web so sémantikou sú dokumenty alebo časti dokumentov reprezentované ontologickými konceptmi. V príspevku opisujeme novú metódu, ktorá realizuje porovnávanie inštancií ontologických konceptov aj s ohľadom na personalizáciu prezentovaných informácií či navigácie v rozsiahlych informačných priestoroch. Pri určovaní podobnosti zohľadňujeme štruktúru ontológie, využívame odlišné stratégie pre výpočet podobnosti dátových a objektových vlastností. Pre potreby personalizácie pátrame po príčinách (atribútoch), ktoré mohli ovplyvniť záujem používateľa o obsah. Navrhujeme spôsob zohľadnenia modelu používateľa s cieľom dosiahnuť presnejšie určenú podobnosť pre jednotlivých používateľov.

Kľúčové slová: ontológia, koncept, inštancia, podobnosť, porovnanie, rekurzia

1 Úvod

V aplikáciách, ktoré prezentujú informácie z rozsiahlych informačných priestorov, akým je napríklad web, alebo podporujú navigáciu v takýchto priestoroch s využitím personalizácie, pristupujeme ku každému používateľovi individuálne s cieľom poskytnúť mu vhodný informačný obsah. Prispôsobovanie rôznych viditeľných aspektov je spravidla založené na charakteristikách, ktoré opisujú vlastnosti používateľa. Tieto sú uložené v modeli používateľa. Aby bolo prispôsobovanie prezentácie informácií a/alebo navigácie čo najlepšie, potrebujeme model používateľa dostatočne naplnený charakteristikami, ktoré sú aktuálne a odrážajú vlastnosti reálneho používateľa v danom čase.

Existuje viacero prístupov, ako získať charakteristiky používateľa pre potreby naplnenia a udržiavania modelu používateľa. Niektoré informácie môžeme získať priamo od používateľa tak, že mu položíme otázku, necháme ho vyplniť formulár

a pod. Menej dôveryhodné sú informácie, ktoré si o používateľovi odvodíme na základe sledovania jeho činnosti [7] či analýzou záznamov webových serverov [1].

Ďalší prístup uvedieme na príklade. Veľa personalizovaných aplikácií ponúka používateľovi možnosť ohodnotiť záujem o obsah, prípadne vyjadriť spokojnosť. Ohodnotenie sa bude pravdepodobne meniť, ako sa bude meniť záujem používateľa o obsah. A teda má zmysel pátrať po dôvodoch jeho ohodnotenia. Predpokladajme, že naším obsahom budú pracovné ponuky z oblasti informačných technológií. Zrejme dokážeme bez veľkých problémov natrafiť na pracovné ponuky, ktoré vyžadujú stredoškolské vzdelanie, minimálne tri roky skúseností, ovládanie základov webových technológií, ponúkajú motivujúce ohodnotenie a pod. Vyberme dve takéto ponuky, ktoré sa budú odlišovať len v mieste výkonu práce. Nech bude miesto výkonu práce jednej ponuky v Európe, napr. v Londýne a druhej ponuky v Spojených štátoch amerických, napr. vo Washingtone, D.C.

Ak uchádzač o zamestnanie priradí týmto dvom pracovným ponukám rôzne hodnoty záujmu, môžeme usúdiť, že záujem bol ovplyvnený práve miestom výkonu práce. Skutočnosť, či chce uchádzač o zamestnanie z Európy pracovať doma v Európe (priradil vyššie hodnotenie záujmu ponuke v Londýne) alebo je to dobrodruh, ktorý chce spoznať iné krajiny (priradil vyššie ohodnotenie ponuke vo Washingtone, D.C.), pre nás nie je na tejto úrovni dôležitá. V tomto prípade nás zaujímala hodnota atribútu, čiže sme usudzovali o *charakteristikách používateľa*.

Ak by náš uchádzač o zamestnanie ohodnotil spomínané pracovné ponuky rovnako, môžeme z toho usúdiť, že miesto výkonu práce (ako atribút pracovnej ponuky) pre neho nie je dôležité. Hovoríme teda o *preferenciách používateľa*. Z uvedeného príkladu vyplýva, že má zmysel pátrať po dôvodoch hodnotenia záujmu o obsah. Teda zdrojom informácií o používateľovi je analýza obsahu, ktorý sa prezentuje používateľovi [6].

V príspevku opisujeme metódu porovnávania inštancií ontologických konceptov s cieľom identifikovať spoločné a odlišné aspekty konceptu pre účely personalizácie. Ilustračné príklady uvádzame z domény pracovných ponúk, ktorá bola rozpracovaná v rámci výskumného projektu NÁZOU[1] [16]. To však neznamená, že navrhnutú metódu možno použiť len pre doménu pracovných ponúk. V nasledujúcej časti uvádzame charakteristiku ontologických konceptov a ich inštancií, čo definuje kontext navrhnutej metódy. V časti 3 opisujeme existujúce prístupy k porovnávaniu inštancií ontologických konceptov. Nasleduje časť venovaná podobnosti a opisu spôsobu jej výpočtu pri porovnávaní inštancií konceptov. Navrhnutá metóda porovnávania rekurzívnym prechádzaním inštancie konceptu je opísaná v časti 5. V časti 6 opisujeme spôsob, akým zisťujeme príčiny, ktoré spôsobili podobnosť/rozdielnosť inštancií konceptov. Na záver, v časti 7, uvádzame zhrnutie základných prínosov metódy, použitie metódy v kontexte ďalších nástrojov, experimentálne overenie a možnosti ďalšej práce.

[1] NÁZOU – Nástroje pre získavanie, organizovanie a udržovanie znalostí v prostredí heterogénnych informačných zdrojov, http://nazou.fiit.stuba.sk

2 Ontologické koncepty a ich inštancie

Jednoduché koncepty sú zložené z atribútov, ktorým môže byť priradená hodnota spravidla definovaná abstraktným dátovým typom. Iniciatíva webu so sémantikou so sebou prináša reprezentáciu ontológiami, ktoré ponúkajú bohatšiu štruktúru. Základ pre ontológie tvoria koncepty, ktoré označujú množiny konkrétnych objektov [19]. Koncepty môžu byť usporiadané v hierarchii.

Na obrázku 1 je znázornený koncept reprezentujúci pracovnú ponuku v ontológii pracovných ponúk. S konceptom súvisia atribúty (relácie), ako počet pracovných hodín v týždni (*jo:hoursPerWeek*), dátum nástupu do práce (atribút *jo:startDate*), typ pracovnej zmluvy (*jo:hasContractType*), pracovná pozícia (*jo:offersPosition*), ponúkaná mzda (*jo:hasSalary*) a pod.

jo:JobOffer		
jo:hoursPerWeek	Float	
jo:text	String*	
jo:subordinateCount	Integer	
jo:startDate	String	
jo:StartDateASAP	Boolean	
jo:hasSalary	Instance	jo:Salary
jo:hasPrerequisite	Instance*	jo:Prerequisite
jo:hasDutyLocation	Instance*	r:Region
jo:hasResponsibility	Instance*	jo:Responsibility
jo:hasBenefit	Instance*	jo:Benefit
jo:isOfferedVia	Instance	jo:Organization
jo:hasApplyInformation	Instance*	jo:ApplyInformation
jo:hasJobTerm	Instance*	jo:JobTerm
jo:hasContractType	Instance*	jo:ContractType
jo:offersPosition	Instance	c:ProfessionClassification
jo:involvesTraveling	Instance	jo:TravelingLevel
jo:isOfferedBy	Instance	jo:Organization
jo:isOfManagementLevel	Instance	jo:ManagementLevel

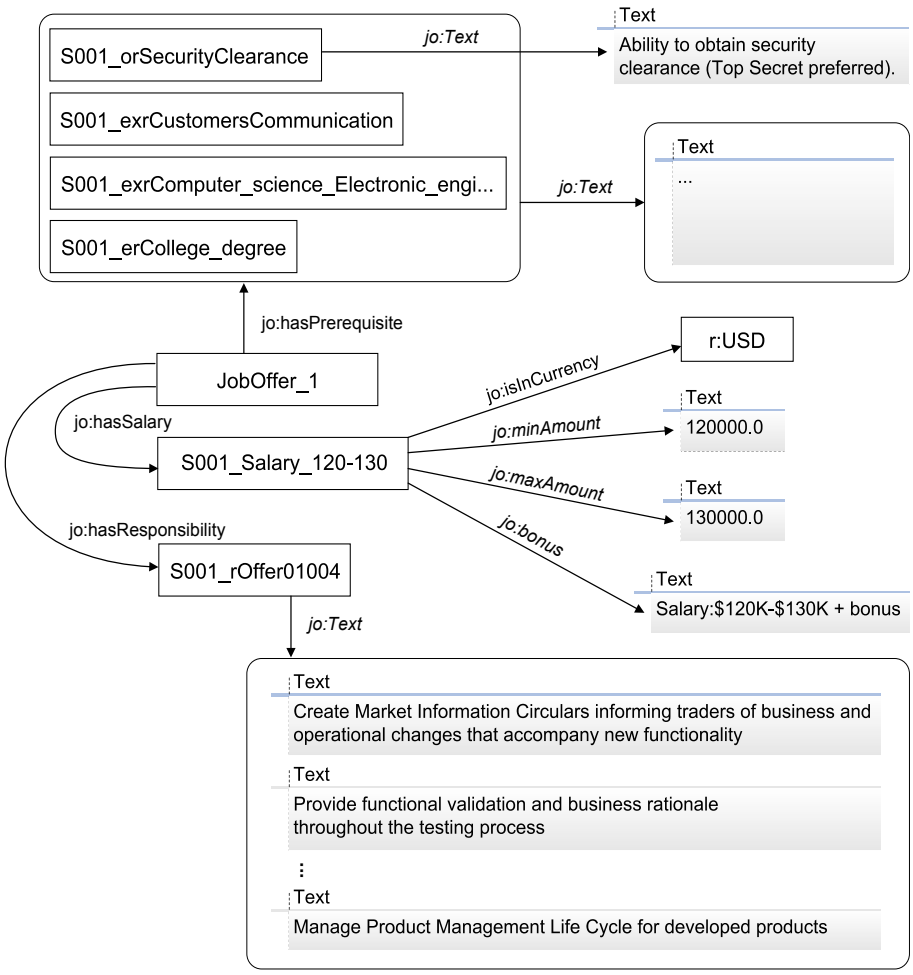
Obrázok 1. Koncept reprezentujúci pracovnú ponuku a súvisiace atribúty.

Vo všeobecnosti ontologické koncepty definujú dátové alebo objektové atribúty. Objektové atribúty sú naviazané na ďalšie koncepty alebo ich inštancie. Zároveň môžu byť niektoré atribúty násobné. Násobné atribúty sú na obrázku 1 symbolizované hviezdičkou (napr. *jo:hasPrerequisite*). Inštancie zodpovedajú objektom reálneho sveta. Príklad konkrétnej ponuky práce je na obrázku 2.

Existujú dva pohľady na koncept [10], t.j. *extencionálny* (angl. extensional) a *intencionálny* (angl. intensional). Pri extencionálnom pohľade je koncept zložený z množiny objektov, ktoré sú inštanciami konceptov domény a množiny

atribútov, ktoré koncept opisujú. Zjednodušene povedané, dáta aj meta-dáta o nich sú uložené v ontológii. Pri intencionálnom pohľade je koncept zložený z množiny atribútov, ktoré ho opisujú. To je typické pre web so sémantikou.

Vzájomným vzťahom medzi extencionálnym a intencionálnym pohľadom sa zaoberá formálna analýza konceptov (angl. Formal Concept Analysis). V poslednom čase sa čoraz častejšie objavujú "extencionálne" ontológie, a preto vzniká potreba pracovať s inštanciami takýchto konceptov.



Obrázok 2. Príklad inštancie pracovnej ponuky. Každý objekt na obrázku má svoj jedinečný identifikátor. Kvôli prehľadnosti uvádzame na obrázku len menovky objektov. Menovky dátových atribútov sú zvýraznené kurzívou. Viacnásobné atribúty (napr. *jo:hasPrerequisity*) sú ohraničené zaokrúhleným rámčekom.

3 Prístupy k porovnávaniu konceptov

Opis metódy na výpočet podobnosti FCA konceptov a identifikáciu konceptov, ktoré sú si sémanticky blízke, využívajúcej intencionálny aj extencionálny pohľad je opísaný v [10]. Koncept je definovaný v kontexte (O, A, R), kde O je množina objektov, A je množina atribútov a R je binárna relácia medzi O a A. Hlavnou nevýhodou tohto prístupu je potreba ontológie podobností obsahujúcej mieru podobnosti medzi entitami doménovej ontológie (napr. podobnosť slov "pláž" a "pobrežie" ohodnotená na 0,9). Navrhnutý prístup umožňuje určiť podobnosť pre viac ako dva koncepty súčasne, prípadne určiť podobnosť konceptov z odlišných kontextov.

Nástroj SymOntos [11] bol vytvorený v rámci európskeho projektu zameraného na vytváranie a udržiavanie ontológií v oblasti cestovného ruchu. Porovnávanie konceptov prebieha v dvoch fázach, pričom dôraz sa kladie najmä na štruktúru konceptu. V prvej fáze predspracovania konceptu sú zostrojené dva grafy – *graf dedičnosti* usporiada koncepty podľa hierarchie zovšeobecnenia a *graf podobnosti*, v ktorom uzly zodpovedajú konceptom a hrany majú priradený stupeň podobnosti. Výpočet podobnosti prebieha v druhej fáze, ktorá je rozdelená do troch krokov. V prvom kroku sú využité ploché štrukturálne vlastnosti (*part*, *related*, *predicate*), nasleduje zohľadnenie hierarchickej štruktúry a v poslednom kroku sa podobnosť vypočíta kombináciou predchádzajúcich krokov.

Princíp porovnávania s ideálnym konceptom (u nás pracovnou ponukou) je použitý v metóde na vyhľadávanie podľa kritérií používateľa, ktorú realizuje nástroj *CriteriaSearch* [17]. Výhodou metódy je, že umožní vyhľadanie aj takých ponúk, ktoré nespĺňajú kritériá ideálnej ponuky úplne. Navyše pri určovaní zhody medzi dvomi ponukami môže používateľ určiť pre každé kritérium, či musí byť splnené, dôležitosť kritéria a presnosť splnenia.

Problém hľadania podobnosti v ontológiách nie je nová myšlienka. Tieto princípy sa využívajú v oblasti mapovania ontológií zameranej na zvýšenie znovupoužitia a spolupráce rozdielnych ontológií opisujúcich tú istú aplikačnú doménu. Napríklad v [20] je opísaný prístup, ktorý využíva inštancie ontologických konceptov ako jednu z heuristík pri identifikácii zmien v ontológiách.

Prístup na určenie podobnosti konceptov opísaný v [18] využíva tri nezávislé ohodnotenia. Na prvej úrovni zohľadňuje synonymické slová. Následne sa pridá sémantika zapracovaním charakteristických čŕt konceptu. Na poslednej úrovni sa použijú sémantické väzby, aby sa zistilo, či prepojené entity spolu súvisia. Nakoniec sa vypočíta vzdialenosť medzi konceptmi metódou najkratšej cesty.

V [15] je opísaný štvorkrokový prístup realizujúci mapovanie medzi ontológiami zahrňujúci podobnosť menoviek (tried, inštancií, vzťahov), inštancií, štruktúry a predchádzajúcich výsledkov mapovania overených aplikáciou. Na porovnávanie inštancií je použitá metóda Edit-Distance v kombinácii s Glue (nástroj využívajúci techniky strojového učenia [9]).

Spoločným znakom opísaných prístupov je, že nepátrajú po dôvodoch, ktoré spôsobili hodnotenie. Náš prístup umožňuje zdôvodnenia, čo prispieva aj k významnej vlastnosti modelu používateľa ako je skúmateľnosť (angl. scrutability) [13].

4 Podobnosť inštancií konceptov

Formálne môžeme podobnosť dvoch objektov x a y zadefinovať takto [4]:

- $sim\,(x, y) \in [0..1]$,
- $sim\,(x, y) = 1 \rightarrow x = y$: dva objekty sú identické,
- $sim\,(x, y) = 0$: dva objekty nemajú nič spoločné,
- $sim\,(x, x) = 1$: podobnosť je reflexívna (objekt je identický sám so sebou),
- $sim\,(x, y) = sim\,(y, x)$: podobnosť je symetrická.

Každé uskutočnené porovnanie prispieva k celkovej podobnosti konceptov, ktorá je daná súčtom týchto čiastkových podobností:

$$celkovaPodobnost = \frac{\sum ciastkovaPodobnost}{pocetPorovnani}, \tag{1}$$

kde *ciastkovaPodobnost* je číslo z intervalu <0,1> a vyjadruje, nakoľko sú porovnávané objekty podobné pri použití konkrétnej stratégie na porovnávanie; *pocetPorovnani* udáva celkový počet porovnaní, ktoré boli uskutočnené.

Pri porovnávaní inštancií ontologických konceptov často nastáva situácia, kedy porovnávané inštancie majú nielen odlišné hodnoty atribútov, prípadne ich počet (ak sú atribúty viacnásobné), ale aj samotné atribúty sú odlišné, keďže inštancia nemusí obsahovať všetky atribúty tak, ako sú definované v koncepte.

V prípade rozdielneho počtu atribútov budeme vo vzťahu pre výpočet celkovej podobnosti uvažovať vždy väčší počet atribútov (počet porovnaní). Ak by sme vo výpočte uvažovali menší počet atribútov a obidva koncepty by sa zhodovali v týchto atribútoch, dostali by sme identitu aj napriek tomu, že jeden z konceptov by mal viac atribútov.

V prípade, že sa nejaký atribút v jednej z inštancií nenachádza, nevieme vyhodnotiť čiastkovú podobnosť, pretože nemáme s čím uskutočniť porovnanie. Podľa [5] sémantická podobnosť je špeciálnym prípadom sémantickej príbuznosti (angl. relatedness), kde príbuznosť pokrýva širšie vzťahy medzi konceptmi a zahŕňa v sebe aj podobnosť. V zmysle tejto definície o podobnosti hovoríme v prípade, že obidva koncepty majú rovnaké atribúty. A teda v tomto špeciálnom prípade by sme mali hovoriť už sémantickej príbuznosti.

Jednoduchým spočítaním čiastkových podobností dostaneme pre dve inštancie vždy rovnaký výsledok v každom kontexte porovnávania (samozrejme pri použití rovnakých stratégií porovnávania). Ak máme k dispozícii model používateľa, zohľadnenie subjektivity pri porovnávaní môže zlepšiť výsledky porovnávania.

Pre potreby personalizácie sme navrhli zavedenie princípu váh do výpočtu podobnosti, čo umožní pre každého používateľa vypočítať podobnosť s ohľadom na jeho jedinečnosť. Zovšeobecnený vzťah pre výpočet celkovej podobnosti je takýto:

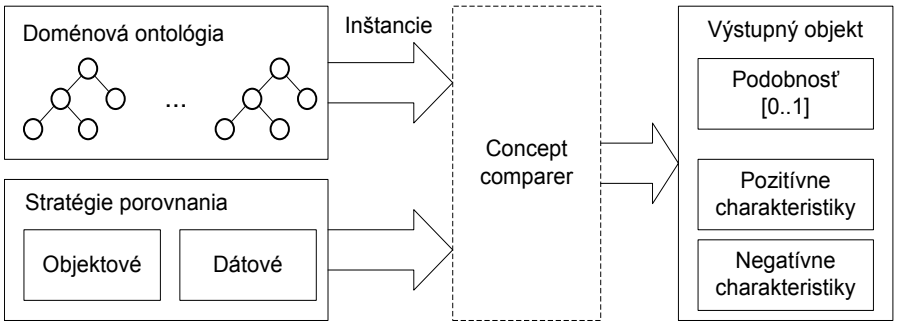
$$celkovaPodobnost = \frac{\sum ciastkovaVaha.ciastkovaPodobnost}{pocetPorovnani},$$

kde význam jednotlivých premenných je rovnaký ako vo vzťahu (1) pre výpočet celkovej podobnosti. Premenná *ciastkovaVaha* nadobúda hodnoty z intervalu <1,2>. To, čo odhadujeme ako vhodné pre používateľa (dané charakteristikami v modeli používateľa), by malo viac zavážiť. V princípe každá čiastková podobnosť má váhu rovnú 1, čo sa nijako neprejavuje vo výpočte. Preto čiastkovú podobnosť atribútu konceptu zvýšime, ak v modeli používateľa existuje zodpovedajúca charakteristika a jej hodnota sa rovná hodnote atribútu konceptu. Návrh zvýšenia váhy tak, aby to bolo dostatočné na ovplyvnenie celkovej podobnosti, treba vykonať experimentálne. My sme v prvotných experimentoch použili dvojnásobnú váhu (hodnota premennej *ciastkovaVaha*).

V prípade, že sa v modeli používateľa nachádza zodpovedajúca charakteristika, ale jej hodnota nie je rovná hodnote atribútu konceptu, váha môže nadobúdať hodnotu z intervalu (1,2).

5 Porovnávanie rekurzívnym prechádzaním inštancie konceptu

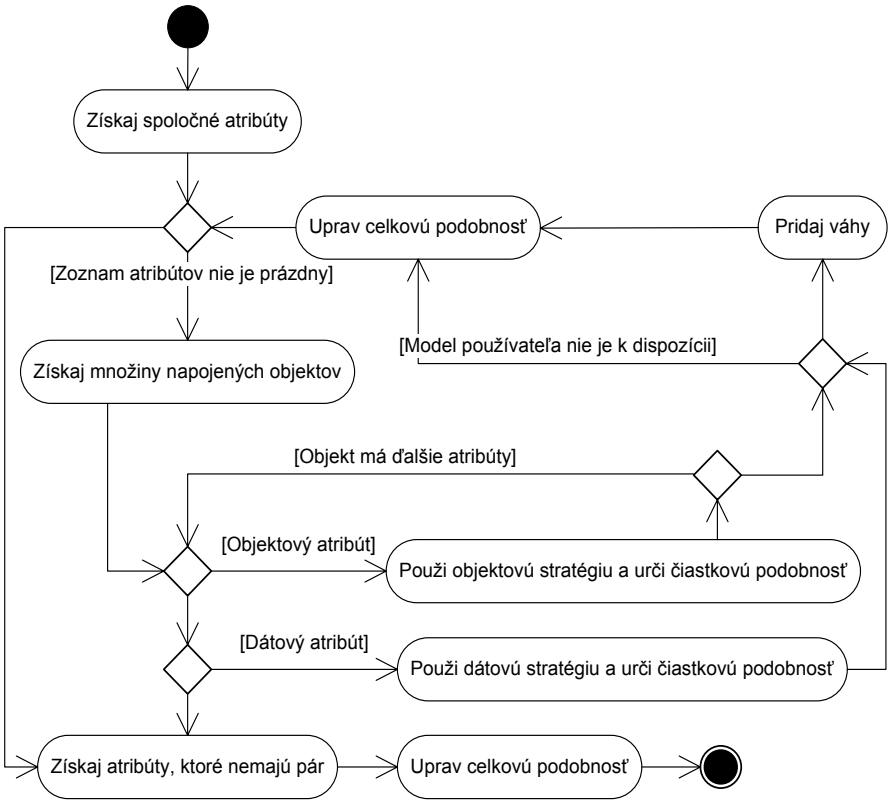
Prehľad vstupov a výstupov metódy je znázornený na obrázku 3. Vstupom sú dve inštancie z tej istej doménovej ontológie, ktoré sa majú porovnať a súbor stratégií pre porovnanie dátových a objektových atribútov. Výstupom je kvantitatívne vyjadrená podobnosť a množina pozitívnych a negatívnych charakteristík (pozri časť 6). *Concept Comparer* je pomenovanie softvérového nástroja, ktorý realizuje porovnávanie inštancií konceptov navrhnutou metódou.



Obrázok 3. Vstupy a výstupy metódy.

Princíp navrhnutej metódy je znázornený na obrázku 4. Vyhodnotenie podobnosti začína získaním všetkých atribútov, ktoré sú spoločné pre obidve inštancie. Ak sa na ontologickú reprezentáciu pozrieme z pohľadu trojíc (t.j. subjekt, predikát a objekt), atribút zodpovedá predikátu. Ako bolo spomenuté vyššie, niektoré atribúty môžu byť viacnásobné. Napr. pracovná ponuka môže mať viacero požiadaviek na predchádzajúce skúsenosti. V tomto kroku získame len jeden výskyt viacnásobného atribútu.

Z množiny získaných atribútov postupne vyhodnotíme všetky atribúty. Vyberieme atribút a v tomto okamihu potrebujeme rozlíšiť, či je atribút objektový alebo dátový. V prípade dátového atribútu použijeme niektorú zo stratégií na vyhodnotenie dátových atribútov a algoritmus končí.



Obrázok 4. Princíp metódy porovnávania prechádzaním inštancie konceptu.

Pre každý objektový atribút získame dve množiny objektov, ktoré sú napojené na atribút (jednu pre každú inštanciu). Ak atribút nie je násobný, množina obsahuje práve jeden objekt. V opačnom prípade treba identifikovať dvojice atribútov, ktoré budeme porovnávať. Rekurzívne prechádzame jednotlivé objekty a skúmame ich vlastnosti (hĺbka rekurzie, počet vlastností, typ vlastností, vzdialenosť v taxonómii, menovky a pod.). Po určení dvojíc použijeme objektové stratégie na určenie podobnosti [2]. Ak spomínané množiny objektov majú rôznu mohutnosť, pre objekty, ku ktorým nemáme dvojicu, použijeme rovnaký prístup, ako sme definovali pri atribútoch. Keďže nemáme s čím porovnať, budeme predpokladať, že objekty sú maximálne odlišné, a teda ich podobnosť je rovná nule.

Počas rekurzívneho prechádzania môžeme natrafiť na inverzné alebo symetrické vlastnosti, ktoré môžu spôsobiť zacyklenie, prípadne prechod na iné inštancie. Napríklad "Washington, D.C." je napojený na pracovnú ponuku atribútom *jo:hasDutyLocation*. Keďže je pravdepodobné, že vo Washingtone, D.C. sa bude nachádzať viac ako jedna ponuka, je vhodné, aby všetky takéto ponuky boli napojené na ten istý objekt. A teda v použitej doménovej ontológii pracovných ponúk existuje inverzná vlastnosť (*jo:isDutyLocationOf*), ktorá toto zabezpečuje. Podobný problém by spôsobili aj vlastnosti, ku ktorým existuje symetrická vlastnosť. Z tohto dôvodu neberieme do úvahy inverzné vlastnosti, ani symetrické vlastnosti k práve prehliadanej vlastnosti.

6 Zisťovanie príčin podobnosti

Cieľom nášho prístupu je nielen určiť podobnosť konceptov, ale zároveň pátrať po dôvodoch, ktoré spôsobili túto podobnosť/rozdielnosť. Z ohodnotenia používateľom, napr. záujmu o koncepty, môžeme odvodiť charakteristiky používateľa. Ak koncept obsahuje atribút, ktorý je pre používateľa dôležitý, pravdepodobne ovplyvní jeho hodnotenie viac smerom k pozitívnym hodnotám a naopak, ak je atribút neprijateľný, smerom k negatívnym hodnotám.

Z tohto dôvodu zavádzame dve prahové hodnoty, ktoré rozdelia atribúty podľa vypočítanej podobnosti na tri množiny. Keďže nás zaujímajú najmä atribúty, ktoré výrazne vplývajú na celkovú podobnosť, prahové hodnoty nerozdelia interval podobnosti <0,1> na rovnaké intervaly. Pre zaradenie do množiny "pozitívnych" atribútov sme navrhli pri experimentoch podobnosť vyššiu ako 0,85 a pre zaradenie medzi "negatívne" atribúty podobnosť menšiu ako 0,15. Overenie prahových hodnôt je predmetom ďalších experimentov. Hodnoty množiny atribútov sa napĺňajú priebežne počas vyhodnocovania podobnosti. Ak hodnota čiastkovej podobnosti spĺňa kritériá pre zaradenie do jednej z množín, pridáme do nej URI atribútu.

Takto získané atribúty môžeme použiť na naplnenie modelu používateľa charakteristikami a tiež na aktualizáciu existujúcich charakteristík v modeli používateľa. Súčasťou opísanej metódy nie je aktualizácia modelu používateľa, ale predpríprava dát pre iné nástroje. Napríklad nástroj *LogAnalyzer* (vytvorený v rámci toho istého výskumného projektu) realizuje automatické získavanie charakteristík používateľa sledovaním správania a analyzovaním záznamov o vykonaných akciách [3]. Úpravou nástroja by bolo možné využitie množiny pozitívnych a negatívnych charakteristík v kombinácii s ohodnotením konceptov spätnou väzbou na naplnenie alebo aktualizáciu modelu používateľa.

7 Experimentálne overenie a záver

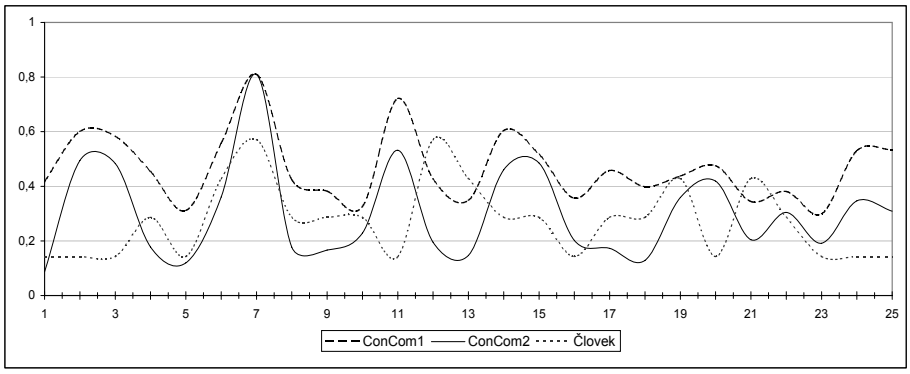
V príspevku opisujeme metódu porovnávania inštancií ontologických konceptov rekurzívnym prechádzaním. Výsledná podobnosť je zložená z čiastkových podobností. Pre potreby personalizácie pátrame po príčinách (atribútoch), ktoré

ovplyvnili používateľov záujem o obsah. Zaviedli sme dve prahové hodnoty, ktoré rozdelili atribúty konceptu podľa vypočítanej podobnosti na tri množiny. Pre potreby personalizácie v aplikáciách webu so sémantikou berieme do úvahy dve krajné množiny (pozitívne a negatívne atribúty). Tieto môžu byť inými nástrojmi transformované na charakteristiky a uložené v modeli používateľa. Navyše, navrhli sme spôsob, ako zohľadniť subjektivitu pri porovnávaní (v prípade, že máme k dispozícii model používateľa), a tým zlepšiť výsledky porovnávania.

Takto vyhodnotená podobnosť je vhodná ako podpora v našom výskumnom projekte NÁZOU pre zhlukovacie algoritmy [12], sémantickú anotáciu [14] a údržbu ontologických úložísk [8]. Metóda je navrhnutá pre prácu s inštanciami z tej istej doménovej ontológie a neuvažuje podobnosť medzi atribútmi. To môže byť ďalším rozšírením metódy.

Overenie opísaných postupov vykonávame vyvinutým softvérovým nástrojom ConCom na doméne pracovných ponúk. Pre ohodnotenie dátových vlastností sme použili Levenshteinovu metódu a na ohodnotenie objektových vlastností sme použili vzdialenosť objektov v taxonómii. Nástroj je implementovaný v jazyku Java. Pri práci s ontologickými modelmi využíva rámec Sesame. Model doménovej ontológie je reprezentovaný v jazyku OWL DL.

Experimenty sa zameriavajú v prvom rade na overenie, či podobnosť získaná softvérovým nástrojom zodpovedá skutočnému stavu. Ako vstup pre overenie sme použili vzorku 300 dvojíc pracovných ponúk, v ktorej sa 30 náhodne vybraných dvojíc nachádzalo dvakrát ako kontrolná vzorka. Dvojice zo vzorky ohodnotil človek na stupnici od 0 do 7, čo sme následne prepočítali na zodpovedajúcu podobnosť z intervalu <0,1>. Výsledky experimentu sú znázornené na obrázku 5.



Obrázok 5. Grafické znázornenie výsledku porovnania pre 25 inštancií pracovných ponúk.

Nástrojom ConCom sme vypočítali dve podobnosti. V prvom prípade boli do celkovej podobnosti zahrnuté len tie atribúty, ktoré sa nachádzali v obidvoch inštanciách (krivka ConCom1) a v druhom prípade sme uvažovali všetky atri-

búty (krivka ConCom2). Ako môžeme vidieť na obrázku 5, krivka 1 nekopíruje úplne tvar krivky 2, keďže jednotlivé inštancie mali rozdielny počet nepárových atribútov. Obidve krivky približne kopírujú krivku, ktorá vznikla ohodnotením pre človeka. Dôvodom pre významnejšie rozdiely môže byť práve jedinečnosť používateľa, ktorý sa rozhoduje na základe niekoľkých atribútov, ktoré sú preňho dôležité. A teda iný používateľ by pravdepodobne podobnosť vyhodnotil ináč. Pri overení kontrolnej vzorky sme zistili, že používateľ pri niektorých rovnakých dvojiciach použil mierne odlišné ohodnotenie podobnosti, čo je dané tým, že používateľ nerozhoduje vždy rovnako.

V ďalšej práci plánujeme experimentovať na vygenerovaných inštanciách, aby sme overili aj časť určenú pre zisťovanie príčin podobnosti a overiť metódu v prepojení s inými nástrojmi, ktoré získané výsledky použijú na naplnenie alebo aktualizáciu charakteristík v modeli používateľa.

Tento príspevok vznikol za podpory Štátneho programu výskumu a vývoja "Budovanie informačnej spoločnosti" č. úlohy 1025/04 a Vedeckej grantovej agentúry VEGA v rámci grantovej úlohy č. VG1/3102/06.

Referencie

1. Andrejko, A., Barla, M., Bieliková, M., Tvarožek, M. User characteristics acquisition from logs with semantics. In: *ISIM '07 Information Systems and Formal Models: 10th International Conference on Information System Implementation and Modeling*, (2007) 103–110
2. Andrejko, A., Barla, M., Tvarožek, M. Comparing ontological concepts to evaluate similarity. In: *Tools for acquisition, organization and presenting of information and knowledge*. Bystrá dolina, Nízke Tatry, Bratislava, (2006) 71–78
3. Barla, M., Bieliková, M. Estimation of User Characteristics using Rule-based Analysis of User Logs. In: *Data Mining for User Modeling Proceedings of Workshop held at the International Conference on User Modeling UM2007*, Corfu, Greece, (2007) 5–14
4. Bisson, G. Why and how to define a similarity measure for object based representation systems. In: *Towards Very Large Knowledge Bases*, (1995) 236–246
5. Budanitsky, A., Hirst, G. Semantic distance in WordNet: An experimental application-oriented evaluation of five measures. In: *Workshop on WordNet and other lexical resources, Second meeting of the North American Chapter of the Association for Computational Linguistics*, Pittsburg, June (2001)
6. Brusilovsky, P., Tasso, C. Preface to Special Issue on User Modeling for Web Information Retrieval. *User Modeling and User-Adapted Interaction.* **14**(2-3), (2004) 147–157
7. Callaway, C., Kuflik, T. Using a domain ontology to mediate between a user model and domain applications. In P. Brusilovsky, C. Callaway. A. Nürnberger, eds.: *Workshop on New Technologies for Personalized Information Access*, Edinburgh, Scotland, UK, (2005) 13–22
8. Ciglan, M., Babík, M., Laclavík, M., Budinská, I., Hluchý, L. Corporate memory: A framework for supporting tools for acquisition, organization and maintenance of information and knowledge. In: *Proceedings of 9th International Conference ISIM'06 "Information Systems Implementation and Modelling"*, Brno, MARQ Ostrava, (2006) 185–192

9. Doan, A. H. et al. Learning to map between ontologies on the semantic web. In: *Proceedings of the 11th international conference on World Wide Web*. Honolulu, Hawaii, USA, ACM Press (2002)
10. Formica, A. Ontology-based concept similarity in Formal Concept Analysis. *Information Sciences*. **176**(18), (2006) 2624–2641
11. Formica, A., Missikoff, M. Concept Similarity in SymOntos: An enterprise management tool. *The computer Journal*. **45**(6), (2002) 583–595
12. Frivolt, G., Pok, O.: Comparison of graph clustering approaches. In Bieliková, M., ed.: *Proceedings in IIT-SRC 2006*, Slovak University of Technology, (2006) 168–175
13. Kay, J. Stereotypes, student models and scrutability. In: Gauthier G., Frasson C., and VanLehn K., eds.: *Proceedings of Fifth International Conference on Intelligent Tutoring Systems*. Springer-Verlag, (2000) 19–29
14. Laclavík, M., Šeleng, M., Gatial, E., Balogh, Z., Hluchý, L. Ontology based text annotation – OnTeA. In Duzi, M., Jaakkola, H., Kangassalo, H., Kiyoki, Y., eds.: *Information Modelling and Knowledge Bases XVIII*. Amsterdam, IOS Press (2006)
15. Liu, X., Wang, Y., Wang, J. Towards a semi-automatic ontology mapping – An approach using instance based learning and logic relation mining. In: *Proceedings of Fifth Mexican International Conference on Artificial Intelligence (MICAI'06)*. IEEE (2006)
16. Návrat, P., Bieliková, M., Rozinajová, V. Acquiring, organising and presenting information and knowledge from the web. In: *CompSysTech'06*. Veliko Turnovo, Bulgaria, Bulgarian Chapter of ACM (2006)
17. Pázman, R. Ontology Search with User Preferences. In: *Tools for Acquisition, Organisation and Presenting of Information and Knowledge*. Bystrá dolina, Nízke Tatry, Bratislava, (2006) 139–147.
18. Rodríguez, M. A., Egenhofer, M. J. Determining semantic similarity among entity classes from different ontologies. *IEEE transactions on knowledge and data engineering*. **15**(2), (2003) 442–456
19. Svátek, V. Ontologies and WWW [in Czech]. In: *Datakon 2002*. Brno, Czech republic, (2002) 1–35
20. Tury, M., Bieliková, M. An approach to detection ontology changes. In: *ICWE '06: Workshop proceedings of the sixth international conference on Web engineering*, Palo Alto, California, ACM Press (2006)

Annotation:

Estimating similarity of the ontological concepts instances for the adaptive application based on Semantic Web

If a user is provided two concepts displaying information content and the user gives her evaluation (i.e. expressing an interest) we can further investigate common and different attribute of the concepts and find out useful information about the user's interests. In this paper we describe a novel method which realizes comparing of ontological concepts instances with regard to personalization of presented information or navigation in large information spaces. While estimating similarity we take into account the ontology structure, exploit variety of strategies for data type and object type attributes similarity. To fulfill the need of personalization we investigate reasons (attributes) that influenced user's interest in the content. We propose an approach to integrate a user model to achieve more accurately computed similarity for particular users.