

OntoCase – prípadové usudzovanie založené na ontológiách

Zoltán Balogh¹, Ivana Budinská¹, Ladislav Hluchý¹

¹ Institute of Informatics, Slovak Academy of Sciences, Bratislava, Slovakia
balogh@savba.sk

Abstract

OntoCase je nástroj, ktorý umožňuje použiť techniky prípadového usudzovania a učenia na základe inštancií nad znalosťami reprezentovanými ontológiami. Cieľom nástroja je odhadnúť výsledok dopytovaného prípadu, charakterizovaného príznakmi ako aj triedou samotného prípadu, na základe minulých prípadov. V rámci OntoCase bola vytvorená ontológia, ktorá umožňuje modelovať prípady, ich kontext ako aj ich výsledky. Nástroj je prispôsobiteľný pre rôzne doménové ontológie. Špeciálnou funkcionalitou nástroja je možnosť špecifikovať tzv. rekurzívne príznaky, ktoré môžu byť použité na získanie hodnôt príznakov rôzne poprepájaných ontológií. S cieľom zlepšiť presnosť odhadu výsledku prípadu používa OntoCase pokročilé metódy na meranie štrukturálnej podobnosti ontologických konceptov a inštancií. Nástroj bol vyvinutý v rámci projektu ŠPVV NAZOU.

1 Úvod

Metóda použitá v nástroji OntoCase (“Ontology-based Case Based Reasoning”) slúži na vyvodzovanie výsledku prípadu na základe podobných prípadov z minulosti. Cieľom metódy je odhadnúť výsledok dopytovaného prípadu a to tak, že sa najprv identifikujú najpodobnejšie prípady uložené v báze prípadov, ktorých výsledok sa použije na odhad výsledku dopytovaného prípadu. Metóda vychádza z klasického prístupu vyvodzovania na základe prípadov (CBR - Case Based Reasoning [2]). CBR sa vo všeobecnosti rieši nasledovnými 4 krokmi:

- výber podobných predchádzajúcich prípadov k dopytovanému prípadu,
- znovu použitie týchto prípadov výberom ich výsledkov alebo riešení,
- adaptácia týchto výsledkov na dopytovaný prípad a
- uloženie dopytovaného prípadu do bázy prípadov po jeho verifikácii.

Štyrmi hlavnými úlohami v rámci CBR sú:

- reprezentácia prípadov,
- výber prípadov,
- adaptácia prípadov a
- učenie sa z nového prípadu a správa bázy prípadov.

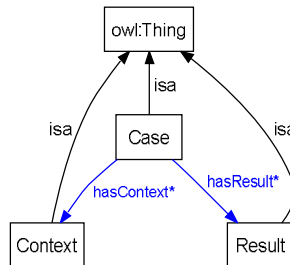
V OntoCase bolo potrebné klasický CBR prístup modifikovať tak, aby dokázal pracovať s ontológiou a to tak, že jednotlivé inštancie tried v ontológii (alebo ich zoskupenia) reprezentujú jednak prípad - resp. jeho kontext, alebo výsledok prípadu.

2 Nástroj OntoCase

Pri klasickom CBR prístupe je prípad popísaný ako množina príznakov a hodnôt a výsledok je popísaný len ako jedna hodnota. Prízny najčastejšie nadobúdajú spojité alebo diskkrétne hodnoty. OntoCase ponúka rámec, ktorý vie uskutočniť prípadové usudzovanie nad ľubovoľnou doménovou ontológiou, samozrejme s vynaložením určitého úsilia na integráciu OntoCase s danou ontológiou. Pre dosiahnutie takéhoto cieľa, museli byť modifikované takmer všetky úlohy definované v rámci CBR (okrem 4.) a to nasledovne:

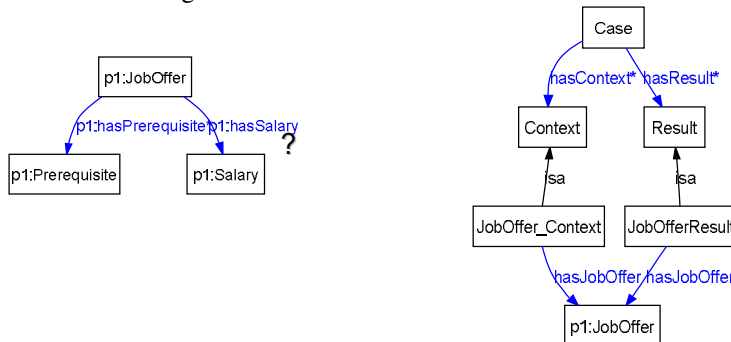
1. Reprezentácia prípadu:

Hlavnou časťou ontológie je reprezentácia prípadu - konceptu Case, ktorý má dve vlastnosti: kontext – koncept Context a výsledok – koncept Result. Model prípadu (Case) je základným bodom celého prístupu OntoCase. Je založený na OWL [3],[4] a rozširuje hlavnú triedu owl:Thing. hasContext sa môže odkazovať na n-inštancií triedy Context a hasResult sa môže odkazovať na ľubovoľný počet inštancií triedy Result. Context je abstraktná trieda a mala by byť rozšírená o koncept, ktorý definuje kontext doménového prípadu.



Obr. 1: Model základného konceptu prípadu (Case).

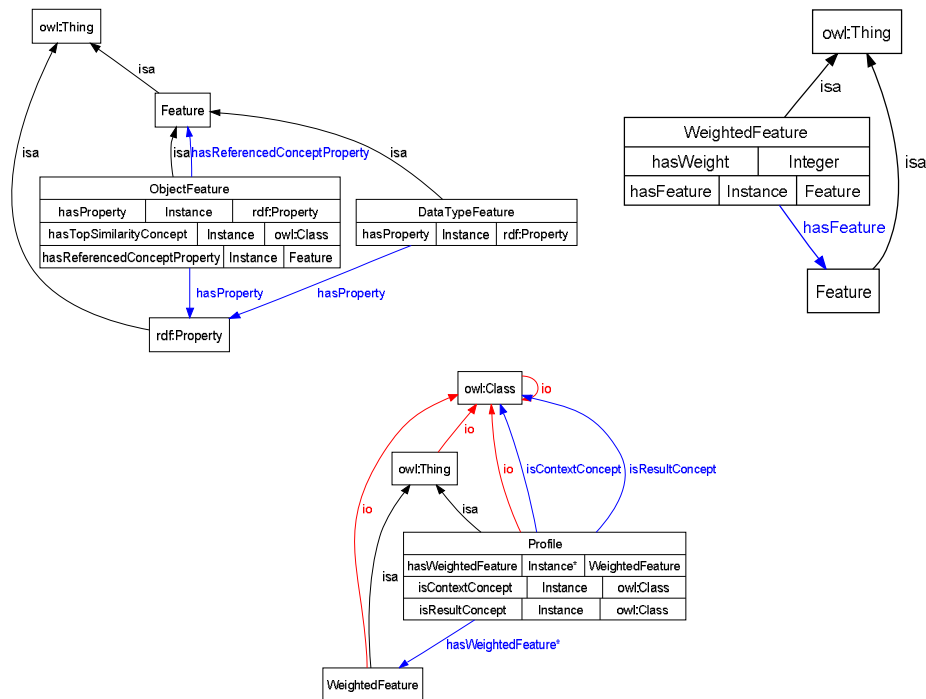
Takýto model prípadu je tak aj základným východiskom pre adaptáciu OntoCase pre rôzne doménové ontológie:



Obr. 2: Príklad napojenia doménovej ontológie.

2. Výber prípadov:

Výber prípadov je navrhnutý s použitím profilov a príznakov – koncepty Profile a Feature. Profile a súvisiace koncepty sa používajú na špecifikáciu príznakov a ich váh počas výberu prípadov. Originality tohto prístupu spočíva v skutočnosti, že príznaky sú dynamicky načítavané priamo z ontológie z vlastností inšancií triedy Resource a jej podtried. Koncept WeightedFeature umožňuje priradiť váhy inštaniciám triedy Feature.



Obr. 3: Model príznaku, vážených príznakov a profilov.

Profile ovplyvňuje spôsob akým sú prípady vyberané. V tejto súvislosti bolo potrebné dosiahnuť niekoľko nasledovných dielčích výsledkov:

- Určenie príznakov z profilu Profile;
- Automatická tvorba SPARQL [5] dotazov určených na získavanie hodnôt príznakov priamo z vlastností inšancií triedy Resource počas výberu prípadov;
- Výber prípadov s kontextom, použitím vygenerovaných SPARQL dotazov. Tu bolo potrebné porovnávať jednotlivé príznaky dotazu s príznakmi prípadov v báze prípadov. Použili sme euklidovskú vzdialenosť na výpočet podobnosti medzi dotazom a prípadmi z bázy prípadov. Z-score sme použili na štandardizáciu hodnôt príznakov. Tiež počítame podobnosť spojitých a diskretných príznakov;
- Vylepšenie výberu prípadov prostredníctvom štruktúrálnej podobnosti konceptov v ontológii. Použitie taxonomicky usporiadaných príznakov umožňuje vý-

počet podobnosti konceptov v ontológii. Použili sme teóriu grafov podobností implementovanú použitím odvodzovania a dopytovania sa v ontológii na výpočet matice podobností.

3. Adaptácia prípadov:

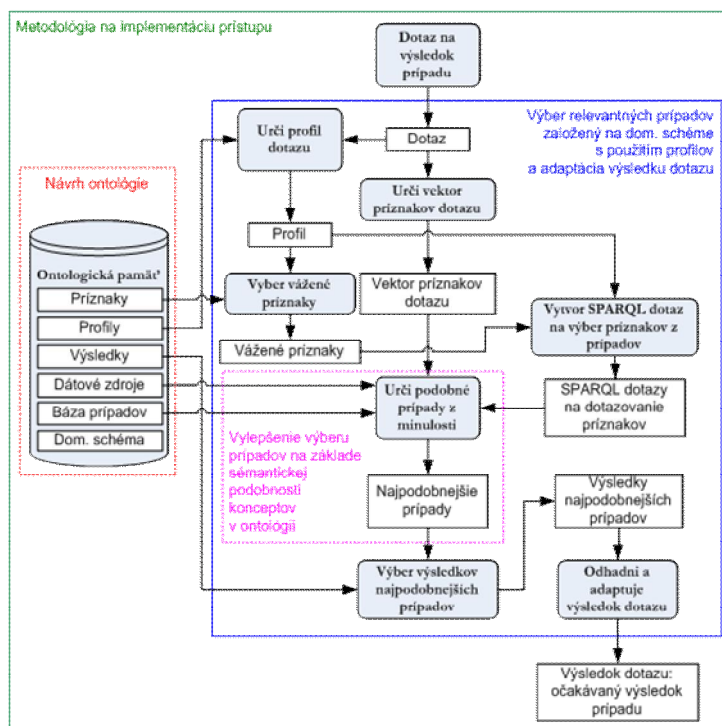
Pre adaptáciu výsledku používame metódu k-najbližších susedov (k-NN), pričom k je dynamicky určované.

4. Správa bázy prípadov:

Napojenie ontológie OntoCase na doménovú ontológiu, ktorá je udržiavaná v rámci ontologickej pamäti, ktorá má vlastné metódy na jej údržbu. Tiež odhadované výsledky dotazov nie sú ukladané späť do korporatívnej pamäte, preto sme sa pri nástroji OntoCase s problematikou správy bázy prípadov nezaoberali.

3 Implementácia

Nástroj je implementovaný pomocou jazyka Java [6] a knižnice Jena Semantic development library [7]. Všeobecná schéma nástroja OntoCase je uvedená na nasledujúcom obrázku.



Obr. 4: Všeobecná schéma nástroja OntoCase.

Biele bloky predstavujú vstupy alebo výstupy jednotlivých fáz prístupu. Bloky s oblými rohmi predstavujú jednotlivé časti predkladaného prístupu. Prerušovanou čiarou sú vyznačené hlavné okruhy práce, ktorými sú:

1. Návrh ontológie – ontologická reprezentácia základných konceptov a vzťahov medzi nimi;
2. Výber relevantných prípadov založený na navrhutej ontológii s použitím profilov a adaptácia výsledku dotazu – v tomto okruhu ide o dekompozíciu dotazu na profil, príznaky a vektor príznakov dotazu, algoritmus na tvorbu SPARQL dotazov používaných na výber vlastností dátových zdrojov a na výber príznakov; výber podobných prípadov k dotazu; výber výsledkov podobných prípadov a odhad a adaptácia výsledku;
3. Vylepšenie výberu prípadov na základe podobnosti konceptov v ontológii je z dôvodu rozsahu samostatným okruhom v rámci problematiky z bodu 2. Tento prístup vylepšuje presnosť celkového procesu výberu prípadov;
4. Metodológia na implementáciu prístupu

4 Záver

Nástroj OntoCase poskytuje odhad výsledku dopytovaného prípadu, charakterizovaného príznakmi ako aj triedou samotného prípadu. Nástroj je konfigurovateľný pre rôzne doménové ontológie. Bola vytvorená metodológia na zavedenie OntoCase do novej doménovej ontológie. Nástroj bol vyvinutý v rámci projektu ŠPVV NAZOU [8] s použitím v doméne pracovných ponúk. Jeho využitie sa plánuje aj v rámci APVV projektu SEMCO-WS [9].

Použitá literatúra

1. Balogh Z.: Knowledge-based Approach to Performance Estimation of Services for Grid Workflows. PhD Dissertation, Institute of Informatics SAS, Bratislava, January 2007.
2. Aamodt. A., Plaza E.: Case-based reasoning: foundational issues, methodological variations, and system approach, AI Communications, vol. 7, no. 1, pp. 39–59, 1994.
3. W3C: OWL Web Ontology Language Guide, W3C Recommendation 10 February 2004. URL: <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>
4. OWL Web Ontology Language Overview, <http://www.w3.org/TR/owl-features/>
5. ARQ - A SPARQL Processor for Jena. URL: <http://jena.sourceforge.net/ARQ/>
6. Sun Microsystems, Inc.: Java Technology, URL: <http://www.sun.com/java/>
7. Jena (A Semantic Web Framework for Java) Homepage. URL: <http://jena.sourceforge.net/>
8. Projekt ŠPVV NAZOU - Nástroje pre získavanie, organizovanie a udržiavanie informácií z heterogénnych zdrojov. URL: <http://nazou.fiit.stuba.sk/>
9. Projekt APVV SEMCO-WS - Semantic composition of Web and Grid Services. URL: <http://semcows.ui.sav.sk/>

Pod'akovanie

Táto práca bola podporená projektmi NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, SEMCO-WS APVV-0391-06, VEGA No. 2/6103/6, VEGA 2/7098/27.