Optimalizace řešení analytických úloh

Semestrální práce na kurz MI-KDD
3. verze (9. listopadu 2015)
Vypracoval: T. Frýda, J. Mráček

Cíl práce

Cílem této semestrální práce je automatizace nalezení požadovaného počtu hypotéz na analytické otázky procedury 4ft-Miner zadané uživatelem.

Vstup

Vstupem programu jsou data a metabáze, nad kterými bude probíhat hledání hypotéz. Uživatel si tak před spuštěním skriptu jednoduše nastaví v LM Workspacu analytické otázky pro 4ft-Miner, ke kterým se budou hledat nějaké hypotézy. U těchto otázek budou uživatelem vyplněné cedenty včetně jejich rozmezí (minimálního a maximálního počtu) a také kvantifikátory včetně jejich hodnot, které budou skriptem použity jako spodní mez *a*. Dalším vstupem bude konfigurační soubor ve formátu lua, který je popsán níže.

Konfigurační soubor

Pro každou automatizovanou analytickou otázku se nastaví:

- maximální počet iterací
- minimum a maximum na očekávaný počet hypotéz
- které kvantifikátory se budou optimalizovat a jak (popis níže)
- omezení výstupu (ano x ne; řadící funkce pro výběr prvních n hypotéz)

Možnosti optimalizace kvantifikátorů

1. **base bude konstantní** a budeme binárním půlením hledat optimum pro PIM nebo AAD (případně další)
2. **ostatní kvantifikátory budou pevně dané** a budeme optimalizovat pouze Base
3. **budeme hledat optimum pro base a jeden kvantifikátor** (PIM, AAD, apod..).
   V tomto případě budeme hledat optimum pro daný kvantifikátor pomocí binárního půlení a Base budeme snižovat o konstantu pokud nenalezneme řešení. Případně naopak.

Algoritmus

Binární půlení

1. provedení nalezení hypotéz tím se získá prahová hodnota a uloží se do proměnné *b*
2. nastavení hodnoty *a* na hodnotu definovanou v konfiguračním souboru (např. 0.4) pokud je vyšší než minimální prahová hodnota v opačném případě nastavíme hodnotu *a* na minimální prahovou hodnotu
3. pokud je pro *b* počet hypotéz větší nebo rovný požadovanému počtu hypotéz končíme
4. pokud se nalezl správný počet hypotéz nebo byl překročen počet iterací končíme
5. nastavíme hodnotu pro hledání hypotéz na *(a+b)/2*
6. pokud je počet hypotéz větší než byl požadovaný nastavíme *a* na *(a+b)/2* v opačném případě nastavíme *b* na *(a+b)/2*
7. přejdeme do kroku č. 4

Výstup

V ideálním případě bude výstupem report s požadovaným počtem hypotéz. Zde se pokusíme zajistit, aby ve výsledku nebyly víckrát obměny jedné hypotézy, což by mohlo uživatele plést - cílem tedy je ve výsledku mít pouze hypotézy logicky nezávislé.

Když se podaří nalézt pouze méně hypotéz, než uživatel požadoval, budou všechny zobrazeny s komentářem, že více se jich nalézt nepodařilo. Naopak při překročení maximálního počtu nalezených hypotéz bude pro jejich zobrazení hrál roli jedna z proměnných v konfiguračním souboru, která určí, jestli si uživatel přeje vypsat všechny, nebo například vybrat pouze ty s nejvyšší mírou spolehlivosti.

Pokud se požadovaný počet hypotéz nalézt nepodaří, bude výstupem chybové hlášení s možnými důvody:

1. příliš vysoká hodnota *a* (dolní mez binárního půlení)
2. hypotézy mohou být u sebe tak blízko, že ani při rozdělení původního intervalu na $2^{počet\ iterací}$ dílů nelze nalézt díl ve kterém byl požadovaný počet hypotéz
3. prahová hodnota *b* byla prvním během nastavena na zbytečně nízkou hodnotu
4. analytická otázka neobsahuje žádnou hypotézu, která by byla pro uživatele zajímavá