

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DOLOVÁNÍ DAT V PROSTŘEDÍ DB SERVERU ORACLE

BAKALÁŘSKÁ PRÁCE

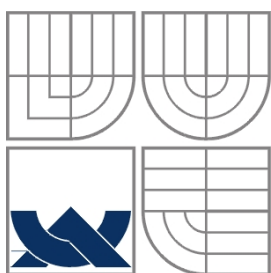
BACHELOR'S THESIS

AUTOR PRÁCE

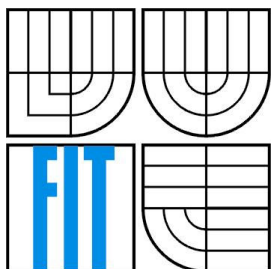
AUTHOR

JURAJ PAP

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

DOLOVÁNÍ DAT V PROSTŘEDÍ DB SERVERU ORACLE

DATA MINING IN ORACLE SERVER ENVIRONMENT

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JURAJ PAP

VEDOUCÍ PRÁCE

SUPERVISOR

Doc. Ing. JAROSLAV ZENDULKA, CSc.

BRNO 2009

Abstrakt

Cílem této práce bylo oboznámit se s problematikou dolování dat a následně zjistit podporu DB serveru Oracle pro dolování. Práce je zaměřená na dolování asociačních pravidel. Výsledkem práce je ukázková aplikace, která umožní uživateli práci s dolovacími modely, změnu jejich nastavení a pomocí grafického rozhraní zobrazení zpracovávaných dat využívajících podporu DB serveru Oracle.

Abstract

The goal of this thesis was to deal with the problem of data mining and finding out support in DB server Oracle subsequently. The project is aimed at mining of association rules. The result of this work is an application that an user can use for working with mining models, changing their options and using an GUI to show the mined association rules. The application uses the support of DB server Oracle.

Klíčová slova

Data mining, získávání znalostí z databáz, jazyk PL/SQL, Oracle Data Mining, asociační pravidla, Oracle Application Express

Keywords

Data mining, knowledge discovery in data, PL/SQL language, Oracle Data Mining, association rules, Oracle Application Express

Citace

Pap Juraj: Dolování dat v prostředí DB serveru Oracle, bakalářská práce, Brno, FIT VUT v Brně, 2009

Dolování dat v prostředí DB serveru Oracle

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením
Doc. Ing. Jaroslava Zendulky, CSc.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Juraj Pap
18. mája 2009

Poděkování

Veľmi rád by som sa poďakoval vedúcemu mojej bakalárskej práce Doc. Ing. Jaroslavovi
Zendulkovi, CSc. za jeho pripomienky a odborné rady. Poďakovanie si zaslúži aj moja rodina. Za ich
trpezlivosť a pomoc. Ďakujem vám všetkým.

© Juraj Pap, 2009

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních
technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je
nezákonné, s výjimkou zákonem definovaných případů..*

Obsah

Obsah.....	1
1 Úvod.....	2
2 Dolovanie dát.....	3
2.1 Čo to je dolovanie dát.....	3
2.2 Proces dolovania dát.....	3
2.3 Druhy dolovacích úloh.....	4
2.3.1 Klasifikácia a predikcia.....	5
2.3.2 Zhluková analýza.....	5
2.3.3 Analýza odlahlých hodnôt.....	5
2.3.4 Popis konceptu/triedy.....	5
2.3.5 Frekventované vzory.....	6
2.3.6 Asociačná analýza.....	6
3 Podpora DB serveru Oracle pre Data mining.....	9
3.1 Oracle Data Mining.....	9
3.1.1 Prehľad funkcií podporovaných ODM.....	9
3.2 Prístup k dolovacím funkciám.....	10
3.2.1 Oracle Data Miner.....	10
3.3 Balíky pre Data Mining.....	10
4 Popis aplikácie.....	14
4.1 Návrh aplikácie.....	14
4.2 Použité technológie.....	14
4.2.1 DB server Oracle.....	14
4.2.2 Oracle Application Express.....	14
4.2.3 SQL*Plus.....	15
4.3 Implementácia aplikácie.....	15
4.3.1 PL/SQL časť.....	16
4.3.2 Grafické rozhranie.....	17
4.3.3 Prepojenie.....	18
4.4 Ukážka aplikácie.....	19
5 Záver.....	24
5.1 Zhodnotenie dosiahnutých výsledkov.....	24
5.2 Diskusia o možnostiach rozšírenia aplikácie.....	24

1 Úvod

V dnešnej modernej dobe, dobe informačných technológií denne rastie množstvo dát skladovaných v databázach. Tie sú preplnené dátami, ktoré na prvý pohľad nič nevypovedajú, ale správnym spracovaním sa môžu poznatky z nich získané stať užitočnými a cennými v rôznych oblastiach spoločenského života. Široká využiteľnosť takto získaných znalostí vo vede, finančníctve a iných odvetviach spôsobila prudký vývoj tejto mladej analytickej metodológie, ktorá dostala názov **Získavanie znalostí v dátach**.

Získavanie znalostí z dát je rýchlo sa rozvíjajúca oblasť. Okrem nových metód a algoritmov sa dnes firmy zameriavajú aj na vývoj aplikácií, ktoré by užívateľom zjednodušili prácu. Preto aj momentálne najväčšia spoločnosť s databázovými systémami na trhu, firma Oracle začala vyvíjať podporu pre svoj databázový systém. Postupným vývojom Oracle DB sa rozširuje aj podpora pre Data Mining. Aktuálne Oracle podporuje všetky základné dolovacie funkcie ako zhľukovanie, regresia, asociačná analýza, dolovanie v texte, porovnávanie sekvencií a ďalšie. Témou tejto práce je oboznámiť sa podrobne s problematikou získavania znalostí z dát a zoznámiť sa s podporou v prostredí databázového serveru Oracle s podporou pre získavanie znalostí. Následne navrhnuť ukážkovú aplikáciu pre implementáciu zvoleného algoritmu dolovania dát. Na záver aplikáciu realizovať a jej funkčnosť otestovať na vhodnej vzorke dát.

Táto práca je zameraná na PL/SQL podporu dolovania dát a ako dolovací model bola zvolená asociačná analýza. Ukážková aplikácia bude vytvorená v aplikácie Oracle Application Express, ktorá je tiež súčasťou DB serveru Oracle a umožní jednoduchý prístup k finálnej aplikácii pomocou webového prehliadača.

Práca pozostáva z úvodu a kapitol, ktoré popisujú jednotlivé etapy riešenia práce.

V druhej kapitole bude čitateľovi priblížená problematika dolovania dát, bude oboznámený s jednotlivými fázami dolovania a bude vysvetlené rozdelenie dolovacích úloh. Podrobnejšie je spracovaná asociačná analýza.

Tretia kapitola sa zaoberá podporou prostredia DB serveru Oracle pre získavanie znalostí. Sú tu vymenované podporované dolovacie úlohy. Popis rozhraní, pomocou ktorých môže čitateľ získať prístup k dolovaniu dát.

Štvrtá kapitola popisuje návrh a následnú implementáciu ukážkovej aplikácie, postup a skúsenosti získane pri práci s jednotlivými fázami dolovania.

Poslednú kapitolu tvorí záver, ktorý obsahuje zhodnotenie práce a výsledkov, diskusiu o možnostiach jej následného rozšírenia aplikácie.

2 Dolovanie dát

2.1 Čo to je dolovanie dát

Dolovanie dát je pojem, ktorý zastrešuje celý proces, pri ktorom sa z dát získavajú netriviálne modely a vzory dát. Dolovanie dát by sa malo správne nazývať **Získavanie znalostí z databáz** (ang. knowledge discovery in data), ale v dnešnej dobe sa pojem dolovanie dát požíva ako alternatíva a dostáva sa stále do popredia. Účelom dolovania dát je pomocou špecifikovaných algoritmov spracovávať dáta spôsobom, na ktorý nepostačujú bežné databázové operácie pomocou SQL príkazov. Nie sú to bežné zostavy ani výkazy. Pri dolovaní sa používa sofistikovaný prístup, s ktorým je potrebné sa oboznámiť. Dolovacie úlohy treba vhodne zvoliť vzhľadom na danú problematiku a očakávané výsledky. Od výsledkov dolovania dát sa očakáva ich novosť a užitočnosť. Za užitočnú informáciu môžeme považovať nájdenú anomáliu v reťazci DNA alebo pri analýze bankových operácií transakcie, ktoré sa líšia natoľko, že sa môže jednať o podvod.

Už pri zmienených príkladoch je badať rôznorodosť využitia v rôznych oblastiach, ktoré od dolovania dát očakávajú výsledky ako efektívnosť zamerania marketingu, zvýšenie tržieb a minimalizáciu strát, odhaľovanie podvodov alebo účinnosť prevencie v medicíne.

2.2 Proces dolovania dát

Proces dolovania dát z databáz je možné rozdeliť na postupnosť krokov, ktoré sa uskutočňujú v opakujúcich sa cykloch

Postupnosť jednotlivých krokov:

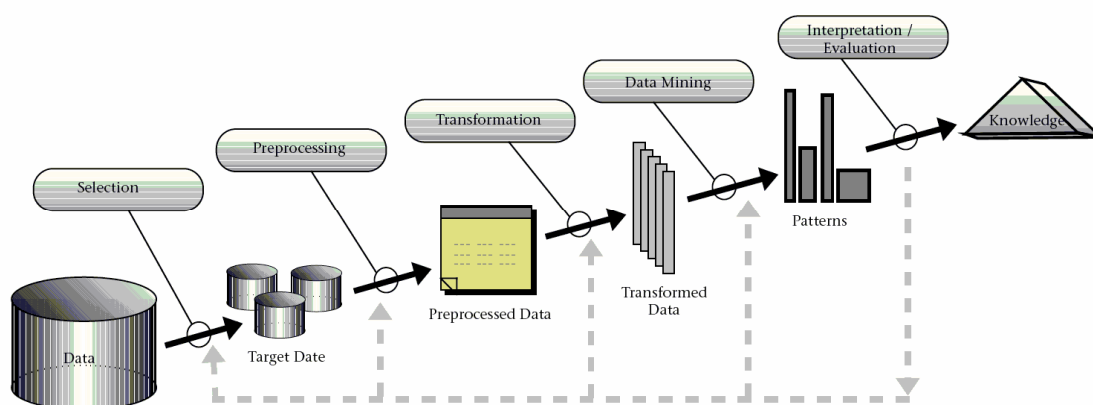
1. Čistenie dát – cieľom čistenia dát je nahradiť chýbajúce hodnoty, vyriešiť nekonzistenciu dát a eliminovať zašumené dáta. Dolovanie nad nečistými dátami môže ovplyvniť spoľahlivosť výsledkov alebo skresliť ich relevantnosť. Napríklad pri zhlukovaní, kde by nulové hodnoty vytvorili triedu, ktorá by bolo považovaná za získane znalosti. Preto je nutné dáta opraviť. Ak je možné chýbajúce hodnoty dopočítať, nahradiť alebo vyradiť z procesu dolovania.
2. Integrácia dát – niekedy je pre dolovanie nutné použiť viacero zdrojov dát, ktoré je potrebné vhodne integrovať. Nie vždy majú rovnakú štruktúru a je potrebné ich vhodne prispôsobiť.
3. Výber dát – výberom sa myslí nastavenie zdroja dát, z ktorého má čerpať.
4. Transformácia dát – prevod dát do formy potrebnej pre zvolený dolovací algoritmus.
5. Dolovanie dát – cieľom je aplikácia algoritmu, ktorého výsledkom budú požadované dáta

6. Hodnotenie získaných modelov a vzorov – pri dolovaní v objemných dátach , dolovací algoritmus môže vrátiť značne veľké množstvo vzorov, pravidiel a dát. Preto je potrebné zhodnotiť ich užitočnosť, novosť a či reprezentujú potrebný výsledok.

7. Prezentácia znalostí – výsledné zistenie treba prezentovať užívateľovi pomocou vhodných prezentačných alebo vizualizačných technik

Z popisu jednotlivých procesov je zrejmé, že prvé štyri kroky slúžia na predprípravu dát, ktorá sa môže líšiť od typu dát. Posledné kroky procesu už vyžadujú interakciu s človekom, ktorý na základe poznatkov o dolovaní a zvolených dátach vyhodnotí užitočnosť získaných znalostí. Výsledné znalosti treba vedieť vhodne prezentovať, aby sa užívateľovi nepredložili iba neprehľadné tabuľky dát, ale ľahko pochopiteľné údaje, ktoré ľahšie spracuje.

Názorná ukážka postupnosti procesu dolovania je vidieť na obrázku 2.1



Obr.2.1: Proces dolovania dát

2.3 Druhy dolovacích úloh

Proces dolovania dát popísaný v predošlej podkapitole je väčšinou totožný pre všetky druhy dolovacích úloh. Hlavná časť procesu sa deje v časti dolovanie dát, kde prebieha vlastná analýza a získavanie vzorov, ktoré sa neskôr môžu transformovať na výsledné znalosti. V tomto bode je potrebné zvoliť vhodnú dolovaciu úlohu a pokiaľ dolovacie prostredie poskytuje viacero algoritmov, je potrebné na základe analýzy dát zvoliť algoritmus, ktorý najlepšie spracuje dáta.

Dolovacie úlohy môžeme rozdeliť na dve základné skupiny: **prediktívne** a **deskriptívne**. Už z názvu jednotlivých úloh je zrejmé, že prediktívne úlohy budú na základe analýzy známych dát vytvárať

vzory, ktoré budú slúžiť ako predpovedanie pre nové dáta. Deskriptívne úlohy charakterizujú dáta podľa ich všeobecných vlastností.

2.3.1 Klasifikácia a predikcia

Tieto úlohy patria medzi prediktívne dolovacie. Klasifikácia je proces, ktorého cieľom je klasifikovanie dát do tried, ktorých je konečný počet, na základe ich vlastností, a tie následne použiť na predikciu triedy objektu, ktorého zaradenie nepoznáme. Rozdiel klasifikácie a predikcie je vo výsledných hodnotách. Výsledné hodnoty klasifikácie sú diskrétné a predikcia vracia hodnoty spojité. Proces predikcie i klasifikácie tvoria tri hlavné kroky: **trénovacia fáza**, **testovacia fáza** a **aplikácia**.

V trénovacej fáze si obe úlohy vytvoria model, ktorý je vytvorený nad trénovacími dátami. To znamená, že u týchto dát poznáme výslednú hodnotu cieľovej premennej. V testovacej fáze sa overuje správnosť vytvoreného modelu pomocou predpovede nad dátami testovacej množiny. Aj u testovacej množiny poznáme cieľovú vlastnosť. Po testovaní treba zhodnotiť úspešnosť modelu. Ak bol model úspešný pre danú predpoveď, môže sa aplikovať na dátach, kde chceme požadovanú vlastnosť predpovedať.

2.3.2 Zhluková analýza

Ako pri klasifikácii aj pri zhlukovej analýze sa analyzované dáta priradujú do tried s tým rozdielom, že pri zhlukovaní sa nepozná priradenie do tried. Naopak zhluková analýza (Cluster analysis) sa zaoberá tým, ako by mali byť objekty (štatistické jednotky) zaradené do skupín tak, aby bola čo najväčšia podobnosť v rámci skupín, a čo najväčšia rozdielnosť medzi skupinami. Takto nájdené triedy majú podobu zhlukov.

2.3.3 Analýza odlahých hodnôt

Pri tejto analýze nehľadáme spoločné znaky prvkov, ani ich nepriradujeme do tried podľa spoločných vlastností. Tento typ analýzy sa špecializuje na vyhľadávanie dát, ktoré sa významne odlišujú od väčšiny. Tento typ úlohy je možné využiť na detekciu podvodov s bankovými kartami alebo transakciami.

2.3.4 Popis konceptu/triedy

Jedná sa o jednu zo základných deskriptívnych dolovacích úloh, kde dáta môžu byť asociované s určitou triedou alebo konceptom. Následne sa vytvorí stručný, súhrnný popis konceptov a tried, ktorý musí byť aj dostatočne presný. Tento popis môžeme získať použitím **charakterizácie dát** alebo **diskriminácie dát**.

Charakterizácia dát – medzi efektívne operácie sumarizácie patrí OLAP operácia roll-up nad dátovou kockou alebo efektívnu techniku generalizácie nad veľkým množstvom dát pomocou metódy **attribute-oriented induction**.

Diskriminácia dát – nepopisuje dáta obecnými výrazmi, ale porovnáva atribúty cieľovej triedy vo vzťahu k jednej alebo viacerým rozdielnym triedam s odpovedajúcimi atribútmi.

2.3.5 Frekventované vzory

Pri odhaľovaní frekventovaných množín v dátach často narazíme na zaujímavé asociácie a korelácie. Frekventované vzory sú často sa opakujúce vzory v dátach. Často súvisia s dolovaním asociačných pravidiel. Vzor môžeme volať frekventovaný, pokiaľ jeho podpora nie je menšia ako stanovená minimálna podpora. Frekventované vzory sa môžu získavať na podstromoch, podgrafoch a podpostupnostiach. Ak sa jedná o kolekcie štruktúr, hovoríme o štruktúrovaných vzoroch.

2.3.6 Asociačná analýza

Pozn.: Táto podkapitola bola prevzatá z [2].

Pôvodná motivácia na získavanie asociačných pravidiel vychádza z potreby analyzovať takzvané transakčné dáta nákupných košíkov, z ktorých sa dá vyvodiť chovanie zákazníkov v zmysle zakúpeného tovaru. Asociačné pravidlá popisujú ako často sú jednotlivé prvky zakúpené dohromady. Ide teda o hľadanie vzájomných väzieb(asociácii) medzi rôznymi položkami sortimentu predajne.

Pritom nie je uprednostňovaný žiadny špeciálny druh tovaru ako záver pravidla.

Výsledné pravidlá môžu byť užitočné pre rozhodovanie o cene produktov, propagácie, vytvárania zásob a mnoho ďalších. Vzorové dáta pre analýzu nákupných košíkov majú formát záznamov jednotlivých nákupov - ID transakcie a výpis zakúpených prvkov. Takto definovaným záznamom hovoríme transakčné dáta.

Získané asociačné pravidlo z takýchto dát by mohlo vyzeráť:

$$\text{“ arašidy} \Rightarrow \text{Coca Cola } (s = 12\%, c = 68\%) \text{”}$$

Pravidlo v takom tvare nám vypovedá, že v 12% nákupov si zákazník kúpil arašidy, a pokiaľ si zakúpil arašidy, bola 68% šanca, že si kúpi aj Coca Colu.

Dve najčastejšie metriky vyjadrujúce významnosť asociačného pravidla je **podpora a spoľahlivosť**.

Podpora(support) vyjadruje percentuálne zastúpenie položiek ľavej strany asociačného pravidla v celkovom počte spracovaných transakcií nákupného košíku. Najčastejšie sa používa ako ohraničujúca hodnota pri niektorých algoritmoch dolovania pravidiel (napríklad Apriori). Definuje sa napríklad podmienka minimálnej podpory - teda získané pravidlá musia mať podporu minimálne `minSupport`.

Spôľahlivosť(confidence) vyjadruje percentuálne zastúpenie položiek pravej strany asociačného pravidla v spracovaných transakciách, kde sa vyskytujú položky z ľavej strany pravidla.

V našom prípade vzorového pravidla bola podpora 12% a spoľahlivosť 68%.

Základná definícia asociačného pravidla

Nech $I = \{i_1, i_2, i_3, \dots\}$ je množina položiek. Ďalej nech D je množina transakcií, kde každá transakcia T je množina položiek taká, že $T \subseteq I$. Ku každej transakcii prislúcha unikátny identifikátor TID . Nech A je množina položiek. Hovoríme, že transakcia T obsahuje A , pokiaľ $A \subseteq T$. Asociačné pravidlo je implikácia tvaru $A \Rightarrow B$, kde $A \subset T, B \subset T$ a $A \cap B = \emptyset$.

Pravidlo $X \Rightarrow Y$ má podporu (*support*) s v množine transakcií D , ak $s\%$ transakcií v D obsahuje množinu položiek $X \cup Y$. Pravidlo $X \Rightarrow Y$ má v množine transakcií spoľahlivosť (*confidence*) c , ak $c\%$ transakcií, ktoré obsahujú X , obsahuje tiež Y .

S využitím pravdepodobnosti možno teda napísať:

$$s(A \Rightarrow B) = P(A \cup B)$$

$$c(A \Rightarrow B) = P(B|A)$$

Proces získavania asociačných pravidiel je tvorený dvomi krokmi:

1. Nájdenie frekventovaných množín, teda množiny položiek splňujúcich podmienku minimálnej podpory
2. Generovanie silných asociačných pravidiel z frekventovaných množín. Tieto pravidlá musia splňovať podmienku minimálnej podpory a spoľahlivosti.

2.3.6.1 Typy asociačných pravidiel

V tejto práci je dôležité spoznať rozdiel medzi rôznymi typmi asociačných pravidiel.

Pravidlá rozdeľujeme do skupín:

Podľa typu hodnôt v pravidlách: Ak nás zaujíma iba prítomnosť alebo neprítomnosť položky, potom sa jedná o booleovské asociačné pravidlá. Ak však pravidlo popisuje asociácie medzi kvantitatívnymi položkami alebo atribútmi, potom hovoríme o kvantitatívnych asociačných pravidlách.

Podľa počtu dimenzií v pravidlách: Booleovské asociačné pravidlá sú jednodimenziálne, obsahujú iba dimenziu *položky transakcie*

Podľa úrovne abstrakcie v pravidlách: Niektoré metódy sú schopné získať asociačné pravidlá nad rôznymi úrovňami abstrakcie. Takýmto pravidlám hovoríme viacúrovňové.

2.3.6.2 Algoritmus Apriori

Slúži k vyhľadávaniu základných asociačných pravidiel a vysvetlíme jeho postup získavania asociačných pravidiel a frekventovaných množín pomocou .

Názov Apriori vyplýva z postupu pri vytváraní frekventovaných množín, pretože pri vytváraní novej množiny sú potrebné frekventované množiny z predošlého kroku. Z toho vyplýva nevýhoda algoritmu, pretože v každom cykle je potrebný prechod databázou.

Apriori postupne generuje množiny frekventovaných položiek, pričom postupuje od najmenších (s najmenším počtom prvkov) k najväčším. Pokiaľ je to možné, z frekventovaných množín s n prvkami sa generujú množiny s $n+1$ prvkami. Súbor frekventovaných množín, ktoré majú n prvkov sa označuje L_n . Výstupom algoritmu sú všetky silné n -množiny, z ktorých je možné generovať asociačné pravidlá. Algoritmus generuje booleovské jednodimenzionálne pravidlá.

Algoritmus používa apriori vlastnosť frekventovaných množín: Všetky neprázdne podmnožiny frekventovanej množiny položiek sú (musia byť) tiež frekventované.

Samotný proces spracovávania dát sa skladá z dvoch krokov **spojovacieho a vylučovacieho**.
Spojovací krok: K nájdeniu L_n (k -všetkých frekventovaných množín), kandidáti na frekventované množiny (C_n) sú vygenerovaný spojením množín z predošlého kroku z L_{n-1} . Nech l_1 a l_2 sú množiny L_{n-1} . Algoritmus predpokladá, že položky v množine sú lexikograficky zoradené. K spojeniu dvoch $(n-1)$ -množín dôjde, pokiaľ ich prvých $n-2$ prvkov je zhodných. Výsledná množina vzniknutá spojením l_1 a l_2 bude mať prvky $l_1[1], l_1[2], \dots, l_1[n-1], l_2[n-1]$, kde $l_1[i]$ je i -tý prvok množiny l_1 a musí platiť, že $l_1[n-1] < l_2[n-1]$.

Vylučovací krok: C_n je nadmnožinou L_n , tzn., že jej prvky buď sú alebo nie sú frekventované. Pri analýze veľkého množstva dát, je neefektívne zisťovanie výskytu každého jedného kandidáta v databáze. Tu sa využíva už spomenutá vlastnosť apriori. Žiadna $(n-1)$ -množina, ktorá nie je frekventovaná, nemôže sa stať súčasťou frekventovanej n -množiny. Takto možno rozpoznať nefrekventované kandidátne n -množiny, pokiaľ aspoň jedna jej $(n-1)$ -podmnožina nie je frekventovaná. Potom túto kandidátnu n -množinu môžeme vylúčiť z C_n .

3 Podpora DB serveru Oracle pre Data mining

3.1 Oracle Data Mining

Podpora pre dolovanie dát bola prvýkrát implementovaná do relačného databázového systému firmy Oracle vo verzii 9iR2 pod názvom Oracle Data Mining(ODM) a je súčasťou Oraclu po aktuálnu verziu. ODM je automaticky inštalovaná vo verzii Oracle Enterprise Edition. ODM poskytuje radu dolovacích algoritmov pre klasifikáciu, regresiu, zhľukovanie, predikciu, asociačnú analýzu, detekciu odľahlých hodnôt, meranie významnosti atribútov. ODM taktiež poskytuje podporu pre predprípravu dát, aby boli vo vhodnom formáte pre dolovacie algoritmy. K dispozícii sú funkcie pre normalizáciu, plnenie, vzorkovanie a ďalšie.

Samotné dolovanie dát bolo navrhnuté, aby dolovacie úlohy prebiehali na DB serveri volaním už zmienených funkcií. Z tejto integrovanej architektúry vyplýva mnoho výhod, ktoré zjednodušujú dolovací proces v prostredí DB servera Oracle a vývoj aplikácií zameraných na získavanie znalostí. ODM sa stáva výkonným a spoľahlivým nástrojom na správu a analýzu dát. Nadbytočný presun dát a špecializované analytické programy sa stávajú nepotrebné. Dolovacie algoritmy operujú nad relačnými tabuľkami alebo pohľadmi v rámci databázy. Následné výsledky je možné zobrazit' pomocou užívateľských aplikácií.

3.1.1 Prehľad funkcií podporovaných ODM

Oracle Data Mining podporuje oba typy získavania znalostí s učiteľom (supervised) a bez učiteľa (unsupervised).

Dolovacie funkcie s učiteľom:

- **Klasifikácia:** k dispozícii sú bayesovská klasifikácia, bayesovské siete, rozhodovacie stromy, Support Vector Machines(SVM) a Logistic Regression
- **Regresia:** Support Vector Machines(SVM) a Linear Regression
- **Meranie významnosti atribútov:** používa sa Minimum Description Length (MDL)

Dolovacie funkcie bez učiteľa:

- **Asociačná analýza:** pre dolovanie asociačných pravidiel a frekventovaných množín je použitý algoritmus A priori

- **Zhlukovanie:** pre spracovávanie dát zhlukovaním slúžia algoritmy k-means a O-Cluster
- **Detekcia odľahlých hodnôt:** One-class Support Vector Machines(SVM)
- **Feature extraction:** Non-Negative Matrix Factorization

Analýza sekvenčných dát: pre prácu s reťazcami DNA slúži algoritmus BLAST

3.2 Prístup k dolovacím funkciám

Oracle podporuje prístup k dolovacím funkciám cez dve programovacie a navzájom kompatibilné rozhrania. Prvé rozhranie PL/SQL API pristupuje k funkciám pomocou balíku DBMS_DATA_MINING a druhé rozhranie je Java API. Java rozhranie je implementáciou Java Data Mining(JDM) 1.0 štandardu. Volanie funkcií a procedúr Oracle PL/SQL API musí byť vhodné prispôsobené, aby sa dalo priamo odkazovať v Jave. Pomocou PL/SQL API je možné volať funkcie a procedúry vo všetkých prostrediach Oracle, ktoré umožňujú prácu s PL/SQL a vnorenými tabuľkami.

Funkčnosť balíku DBMS_DATA_MINING je popísaná v kapitole 3.3.

3.2.1 Oracle Data Miner

Od Oracle verzie 10R1 umožňuje využívať grafické užívateľské rozhranie Oracle Data Miner, ktoré poskytuje jednoduchú manipuláciu s dátami a následne ich spracovanie.

Získavanie znalostí zjednodušuje Oracle Data Miner sprievodca, ktorý sprevádza užívateľa počas jednotlivých fáz procesu získavania znalostí ako predpríprava dát, vytvorenie modelu a následného vyhodnotenia výsledkov. Pre aplikáciu vytvoreného modelu mimo Oracle Data Miner slúži generátor PL/SQL kódov, ktoré obsahujú všetky fázy procesu dolovania.

Následne sa môžu tieto získané kódy využiť v užívateľských aplikáciách, pretože Oracle Data Miner používa funkcie pre Oracle Data Mining.

3.3 Balíky pre Data Mining

Jednotlivivé časti procesu získavania znalostí zabezpečujú funkcie a procedúry obsiahnuté v Oracle systémových balíkoch **DBMS_DATA_MINING** a **DBMS_DATA_MINING_TRANSFORM**

Balík **DBMS_DATA_MINING_TRANSFORM** je určený na fázu predprípravy dát. Poskytuje niekoľko utilít pre prípravu dát. Využívanie funkcií tohto balíku nie je nevyhnutné pre dolovanie dát. Pre prípravu dát môžeme využiť vlastné prostriedky alebo prostriedky tretích strán. Vstupy a výstupy do funkcií obsiahnutých v tomto balíku sú jednoduché tabuľky a pohľady, ktoré nemajú špeciálny

Oracle formát(Oracle proprietary), pretože samotné SQL obsahuje mnoho dostatočne silných možností ako zaobstarať transformácie potrebné pre prípravu dát dostatočne efektívne. Avšak dotazy prevádzajúce transformácie môžu byť miestami prídlhé.

Funkcie na predprípravu sa rozdeľujú na 2 skupiny:

Transformačné metódy- podporuje metódy ako Normalizácia, Binning(rozdelenie definičného oboru na intervaly) a nahradzovanie chýbajúcich hodnôt.

Kroky v definícii transformácie- DBMS_DATA_MINING_TRANSFORM podporuje programy, ktoré definujú CREATE, INSERT a XFORM operácie na definovanie transformácie dát a uskutočnenie nasledujúcich krokov.

Následne je v Oracle dokumentácii popísaný popis jednotlivých podporovaných funkcií a programov na transformáciu dát.

Balík **DBMS_DATA_MINING** zabezpečuje funkcionality pre dolovanie dát v Databáze Oracle. Spôsob spracovania je založený na tom že dáta nikdy neopustia databázu a všetko prebieha v databáze pomocou implementovaných funkcií.

Funkcionalita je založená na vytváraní modelov podľa voľby dolovacej úlohy, na vytvorenie modelu slúži funkcia CREATE_MODEL:

```
DBMS_DATA_MINING.CREATE_MODEL (
    model_name           IN VARCHAR2,
    mining_function       IN VARCHAR2,
    data_table_name       IN VARCHAR2,
    case_id_column_name   IN VARCHAR2,
    target_column_name     IN VARCHAR2 DEFAULT NULL,
    settings_table_name   IN VARCHAR2 DEFAULT NULL,
    data_schema_name      IN VARCHAR2 DEFAULT NULL,
    settings_schema_name  IN VARCHAR2 DEFAULT NULL);
```

Premenné ktoré treba definovať je názov modelu, dolovacia funkcia, zdroj dát pre dolovanie a názov stĺpca, ktorý obsahuje identifikátor. Výpis dolovacích funkcií association, attribute_importance, classification, clustering, feature_extraction a regression.

Voliteľný parameter je názov tabuľky obsahujúce nastavenia potrebné pre tvorbu modelu. Ak nie je zadané použijú sa defaultné hodnoty, ak ich algoritmus vyžaduje. Ďalšie voliteľné parametre je vidieť v syntaxi CREATE_MODEL.

Po vytvorení je možné zobrazit' nastavenie modelu pomocou funkcie GET_MODEL_SETTINGS.

Vstupný parameter je názov modelu.

Pre manipuláciu s modelom sú k dispozícii funkcie DROP_MODEL, EXPORT_MODEL a IMPORT_MODEL.

Funkcia DROP_MODEL sa používa aj pri predzakladaní modelu a by sa z databázy odstránil pôvodný model zo starými nastaveniami.

Pri vytvorení modelu pre asociačné pravidlá je možné použiť dve funkcie `GET_FREQUENT_ITEMSETS` a `GET_ASSOCIATION_RULES`.

Funkcia **`GET_FREQUENT_ITEMSETS`** vracia zoznam prvkov typu `DM_ITEMSET`, ktoré obsahujú nasledovné stĺpce: `itemsets_id`, `items`, `support` a `number_of_items`.

Základný dotaz na vypísanie frekventovaných množín:

```
SELECT itemset_id, items, support, number_of_items
FROM
TABLE(DBMS_DATA_MINING.GET_FREQUENT_ITEMSETS('model_name', num_o
f_items));
```

Výsledok takého dotazu je možné vidieť na obr. 4.3.

Funkcia **`GET_ASSOCIATION_RULES`** zoznam asociačných pravidiel zobrazených podľa zadanych vstupných parametrov. Podporuje jediný algoritmus a to Apriori.

Syntax:

```
DBMS_DATA_MINING.GET_ASSOCIATION_RULES (
    model_name          IN VARCHAR2,
    topn                 IN NUMBER DEFAULT NULL,
    rule_id              IN INTEGER DEFAULT NULL,
    min_confidence       IN NUMBER DEFAULT NULL,
    min_support          IN NUMBER DEFAULT NULL,
    max_rule_length      IN INTEGER DEFAULT NULL,
    min_rule_length      IN INTEGER DEFAULT NULL,
    sort_order           IN DMSYS.ORA_MINING_VARCHAR2_NT DEFAULT NULL,
    antecedent_items     IN DYSYS.ORA_MINING_VARCHAR2_NT DEFAULT NULL,
    consequent_items     IN DYSYS.ORA_MINING_VARCHAR2_NT DEFAULT NULL)
RETURN DM_RULES PIPELINED;
```

Pri volaní sa odkazujeme na vytvorený model. Následne parametre slúžia na filtrovanie požadovaných pravidiel. Filtrovať sa dá n-požadovaných pravidiel, zvolené pravidlo, pravidlá s podporou a spoľahlivosťou väčšou ako parameter, pravidlá s požadovanou dĺžkou a ešte je možnosť voliť si usporiadanie výpisu podľa zadanej hodnoty. Dajú sa zobrazit' aj pravidlá s požadovanými hodnotami na ľavej alebo pravej strane pravidla. V ukážkovej aplikácii sú implementované len niektoré filtrovacie možnosti. Výsledok takého dotazu je možné vidieť na obr. 4.4. a 4.5.

Funkcia vracia prvky typu Oracle dátového typu `DM_RULES`.

```
DM_RULES
(
    rule_id              INTEGER,
    antecedent           DM_PREDICATES,
    consequent           DM_PREDICATES,
    rule_support         NUMBER,
    rule_confidence      NUMBER,
    antecedent_support   NUMBER,
    consequent_support   NUMBER,
    number_of_items      INTEGER )
```


Na základe výstupu je možné zobrazit' číslo pravidla, podporu a spoľahlivosť pravidla, podporu ľavej alebo pravej strany pravidla, počet prvkov v pravidle, a samotné asociačné pravidlo, ktoré ma definovaný vlastný dátový typ DM_PREDICATES. Príklad získavania asociačných pravidiel je v baliku ODM_associations.

Oracle podporuje pre dolovanie asociačných pravidiel jediný možný vstupný formát. Može sa jednať o tabuľku alebo view obsahujúce transakčné dáta. Ak je tabuľka vo formáte relačnej, je možné ju pretransformovať pomocou vytvorenia pomocného pohľadu.

Tento vzor som použil na sformátovanie relačnej tabuľky obsahujúcej dáta vo forme číslo transakcie, ID prvku a počet zakúpených prvkov za transakciu. Transakčnosť sa dosiahla použitím Oracle datového typu DM_NESTED_NUMERICAL, ktorý prezentuje atribúty numerických hodnôt. Existuje ešte DM_NESTED_CATEGORICAL, ktorý reprezentuje hodnoty kategorické.

```
CREATE VIEW ODM_market_nested AS
SELECT    t_id,
          CAST(COLLECT(DM_NESTED_NUMERICAL(t_prod_id, t_pocet))
              AS DM_NESTED_NUMERICALS) ITEMS_NESTED
FROM ODM_market_nested
GROUP BY t_id;
```

Oracle poskytuje podobnú podporu pre ostatné data mining úlohy a pre niektoré aj širší výber algoritmov. Používanie niektorých funkcií sa líši od mnou používanej GET_ASSOCIATION_RULES, pretože niektoré modely nepotrebujú pri vytváraní vstupné parametre, sú to tzv. prediktívne ktoré si ich získajú tréningových množín alebo zhukovanie, ktorého úlohou je vytvárať triedy podľa hodnôt jednotlivých prvkov.

4 Popis aplikácie

Vytvorte aplikáciu, ktorá umožní užívateľovi dolovanie dát z tabuliek jemu prístupných a následné zobrazenie výsledkov. Predpokladá sa, že aplikáciu bude používať užívateľ čiastočne oboznámený s problematikou dolovania asociačných pravidiel.

4.1 Návrh aplikácie

V ukážkovej aplikácii som sa zameral na dolovanie asociačných pravidiel a frekventovaných množín pomocou PL/SQL funkcií už implementovaných v Oracli. Následne bolo potrebné si zvoliť spôsob vytvorenia grafického rozhrania, ktoré by užívateľovi umožňovalo voľbu dát, nastavenie dolovacích parametrov a filtráciu získaných asociačných pravidiel podľa potreby. Grafické rozhranie som sa rozhodol vytvoriť pomocou aplikácie Oracle Application Express. Výhodou tejto voľby je, že celý proces dolovania a aj beh aplikácie beží na serveri. K prístupu k výslednej aplikácii je potrebný iba webový prehliadač a správne užívateľské oprávnenia pre daného užívateľa.

4.2 Použité technológie

Oracle ponúka široké možnosti pre vývoj aplikácii pracujúcimi s databázami, dátovými skladmi a ďalšie, využívajúc rôzne rozhrania a programovacie jazyky. Tu je prehľad technológií využitých pri vytváraní aplikácie.

4.2.1 DB server Oracle

Pre prácu s podporou Oraclu bolo potrebné mať prístup k databázovému serveru a mať priradené dostatočné užívateľské práva pre prácu s dátami a balíkmi pre Data Mining. Bolo možné si nainštalovať vlastný Oracle server a priradenie práv alebo použiť školský databázový server využívaný pre výučbu s potrebnou podporou. Ja som si zvolil použiť školské databázové servery a počas práce som nenarazil na závažný problém.

4.2.2 Oracle Application Express

Oracle Application Express(APEX) je vývojový nástroj úzko integrovaný s Oracle databázového serveru verzie 11g. Je určený na vývoj bezpečných webových aplikácií pre prácu s dátami a je prístupná cez ľubovoľný webový prehliadač. Nie je závislá na klientskom operačnom systéme ani

d'alšej Oracle podpore na klientskej strane. Inštalácia aplikácii prebieha iba raz a to na serveri. Výsledné webové aplikácie sú spustené na APEXe a javia sa ako bežná internetová stránka. Je zdarma súčasťou všetkých edícií Oracle Database vrátane základnej Express Edition.

4.2.3 SQL*Plus

Pre komunikáciu so serverom som používal konzolu SQL*Plus, čo malo svoje výhody aj nevýhody. SQL*Plus podporuje plnú implementáciu Oracle SQL, PL/SQL a širokú radu ďalších rozšírení. Práca s konzolou je jednoduchá, ale prostredie sa môže zdať neprehľadné, napr. pri používaní príkazu SELECT, keď jeho výsledok majú byť desiatky až stovky záznamov. Konzola nie je rozdelená na časti, príkazy a výsledky sa zobrazujú po sebe. Niekedy sa užívateľ musí prehrabávať v kóde, aby zistil, aký príkaz použil dva či tri kroky späť, lebo aplikácia neposkytuje históriu zadaných príkazov. SQL*Plus som využil hlavne pre jednoduchú manipuláciu s balíkmi a ich predzakladaním. Volanie procedúr a ich testovanie pomocou anonymných blokov poskytuje rýchlu odozvu. Výhodou bola aj schopnosť konzoly práce s Oracle dátovým typom NESTED_TABLE.

4.3 Implementácia aplikácie

Ukážková aplikácia pozostáva z dvoch hlavných programových častí. Prvá hlavná časť je tvorená balíkom PL/SQL procedúr a funkcií, ktoré majú za úlohu meniť inicializačné nastavenia modelu a jeho následné vytvorenie. Druhá časť tvorí grafické rozhranie, ktoré sa odkazuje na vytvorené funkcie ovládania modelu. Implementáciu bolo potrebné rozdeliť na tieto dve základné časti, pretože pomocou samotného PL/SQL nie je možné vytvoriť užívateľsky prívetivú aplikáciu a nie je vhodné aby dolovanie prebiehalo pomocou konzoly, kde nie je zabezpečené prijateľné formátovanie výstupov na zadané dotazy. Vývojové prostredie Application Express nepodporuje niektoré Oracle systémové typy používané pri dolovaní dát. APEX si nevedel poradiť s typmi typu NESTED_TABLE, DM_NESTED_NUMERICAL a DM_ITEM.

Pri implementácii bolo potrebné riešiť niekoľko menších problémov, ktoré boli nakoniec programovo vyriešené. Jedným z prvých problémov bola používanie funkcie na vytvorenie modelu, ktorá mala viesť spracovať aj chybný vstup dát a reagovať nevytvořením modelu. Problém vznikol pri zaradení funkčnosti, ktorá poskytuje užívateľovi na výber potenciálne vhodné tabuľky alebo pohľady, ktoré obsahujú aspoň v jednom stĺpci Oracle dátový typ DM_NESTED_NUMERICAL, pre zdroj dát modelu. Nedá sa zabezpečiť prístup užívateľov k aplikácii takých, ktorý sú 100% zbehlí v problematike dolovania dát. Druhý problém tvorí fakt, že pokiaľ samotná funkcia na vytvorenie modelu bola spustená s nesprávnymi parametrami vznikla chyba, ktorá nenávratne zablokovala aplikáciu. Následne nefungovala postupnosť príkazov DROP a CREATE MODEL. Okrem iného v tabuľkovom priestore vznikli tabuľky neštandardných názvov, ktoré bolo potrebné vždy ručne odstrániť. Tento problém bol následne ošetrení vhodnou podmienkou v klauzule výnimiek. Výsledná

aplikácia bola testovaná na vytváranie modelu pomocou nesprávnych údajov a je schopná reagovať na takýto nekorektný vstup.

Druhý problém vznikol pri ladení procedúr obsahujúcich dynamické skripty na vytvorenie view. Aj keď používaný reťazec v príkaze EXECUTE IMMEDIATE sa zdal byť správny, konzola vždy vracala chybu pri priradovaní reťazca k príkazu. Tento problém bol pracne vyriešený postupným testovaním príkazu EXECUTE IMMEDIATE od jednoduchých dotazov „select 1 from dual” po zložité. Riešenie problému prepojenia databáz je popísané v podkapitole 4.3.3 Prepojenie.

Po vyriešení alebo obídení vzniknutých problémov bol už samotný postup vytvárania aplikácie bez väčších problémov. Postupné vytvorenie procedúr a funkcií na nastavovanie modelu a následne jeho vytvorenie. Používanie dynamických príkazov na editáciu view. Pre vytvorenie grafickej časti bola dôležitá funkčnosť dynamickej editácie view, ktorá je ovládaná užívateľom, voľbou tabuliek pre dolovanie a pre prepájanie tabuliek.

Poslednou časťou bolo testovanie a zprehľadnenie grafickej časti aplikácie, pre jednoduchšie ovládanie a zabránenie vzniku chýb, ktoré by spôsobovali nenávratné havarovanie aplikácie.

Pri testovaní som narazil na situáciu, keď som sa snažil nastaviť hodnoty podpory pod 1% a dĺžku pravidla viac ako 3. Toto je situácia na ktorú treba hľadiť z dvoch uhlov. Po prvé algoritmus apriori používa hodnotu minimálnej podpory ako prahovú, pre generovanie frekventovaných množín. Pri dolovaní znalostí na rozsiahlymi databázami je dobré mať prehľad, aké nastavenia budú vhodné pre zvolený model a dáta. V tejto testovacej situácii nastal problém príliš dlhého pravidla a malej prahovej hranice na generovanie frekventovaných množín, čo výrazne zaťažilo výkon serveru. Následne odozva aplikácie bola príliš dlhá, a pri pokusoch o refresh alebo prepnutí na inú stránku aplikácie som spôsobil nekorektné založenie modelu. Z toho vyplýva čím nižšia hodnota podpory a spoľahlivosti a čím dlhšie generované pravidlo, tým dlhšie trvá generovanie modelu. S týmto javom treba počítať pri obsluhu aplikácie.

4.3.1 PL/SQL časť

PL/SQL časť je tvorená balíkom ODM_associations, v ktorom sú implementované jednotlivé funkcie na manipuláciu a nastavovanie databázových dolovacích modelov. PL/SQL príkazmi sa dajú data mining modely vytvárať a ovládať pomocou konzoly. Aby bolo možné modely ovládať a pracovať s nimi pomocou aplikácie, bolo potrebné funkcie a procedúry spraviť obecné, aby fungovali pre ľubovoľnú užívateľskú tabuľku, ktorá je vo formáte potrebnom pre dolovanie dát.

Balík obsahuje nasledujúce funkcie a procedúry:

`model_settings` – funkcia slúži na updatovanie hodnôt v tabuľke nastavení. Parametre, s ktorými je funkcia volaná, reprezentujú podporu, spoľahlivosť a maximálnu dĺžku dolovacieho pravidla.

`create_model` – procedúra na vytváranie dolovacieho modelu. Používa funkcie `DROP` a `CREATE Model`.

`edit_view` – procedúra, ktorá pomocou dynamického PL/SQL generuje view, obsahujúce asociačné pravidlá. Pri generovaní sa rozhoduje podľa parametrov, s ktorými sa procedúra volá. Ak neobsahujú prázdne reťazce, tak sa vo view nahradia identifikátory prvkov, ich skutočnými hodnotami. Ak neexistuje alebo nie je zvolená žiadna tabuľka na prepojenie, zobrazí sa iba identifikátor.

`edit_view_freq` – má podobnú funkčnosť ako `edit_view`, s tým rozdielom, že dynamicky generuje view, podľa rovnakých kritérií, nad tabuľkou frekventovaných množín vygenerovaných modelom.

`get_antecedent` – funkcia, ktorej výstupom sú pospájané reťazce ľavých strán asociačného pravidla do jedného. Využíva sa kvôli lepšiemu zobrazeniu asociačných pravidiel v tabuľke. Ak obsahuje pravidlo viacej prvkov na ľavej strane, potom nesformátovaná tabuľka asociačných pravidiel obsahuje toľko záznamov pre dané pravidlo, koľko prvkov obsahuje pravidlo na ľavej strane.

`model_exist` – funkcia slúži na ošetrenie funkčnosti aplikácie. Pri nevhodnej obsluhu aplikácie môže nastať situácia, kedy sa nevytvorí model a následne už definované pohľady nad modelom nemajú zdroj dát. Ak model neexistuje, užívateľovi nie je zobrazený výsledok dotazu. V tejto funkcii je použitý systémový view `DM_USER_MODELS`, ktorý zobrazuje všetky dolovacie modely a ich vlastnosti. Názov tohto view sa líši vzhľadom na verziu Oracle databáze. V Oracle 10gR2 je použitý `DM_USER_MODELS` a v Oracle 11g je pomenovaný `ALL_MINING_MODELS`.

4.3.2 Grafické rozhranie

Grafická časť bola vytvorená pomocou Oracle Application Express(APEX). Toto rozhranie bolo vybrané pre jeho jednoduchú prístupnosť. Pre aplikácie pracujúce s databázami, je potrebné aby server bol dostupný nepretržite, a tým je zabezpečená dostupnosť aj aplikácii fungujúcej nad APEXom. Jej účelom bolo prepojiť interakcie užívateľa s procedúrami a funkciami na obsluhu serveru.

Síce aplikácia umožňuje zobrazovať dáta v reportoch, tabuľkách vracajúcich výsledky dotazov, pomocou SQL aj PL/SQL dotazov, ale nevedela pracovať s typom `DM_PREDICATES`, ktorý bol základom pri zobrazovaní výsledných asociačných pravidiel. Pre správne zobrazenie je používaný pohľad, vytváraný dynamickou procedúrou `edit_view`. Volanie procedúr je zabezpečené procesmi, ktoré sa v tomto prípade aktivujú po stlačení tlačidla. Využívané select listy, ktoré slúžia na voľbu tabliek, sú generované ako dynamické zoznamy ťahajúce dáta zo systémových view Oraclu.

Ovládanie aplikácie je rozdelené do troch samostatných stránok: **nastavenie modelu**, **frekventované množiny** a **asociačné pravidlá**.

Nastavenie modelu

Na tejto stránke je umožnené nastavovanie modelu pred dolovaním, zvolenie základných parametrov a voľba zdroju dát pre dolovací algoritmus. Toto nastavenie môže výrazne ovplyvniť dobu trvania algoritmu.

Frekventované množiny:

Keďže súčasťou dolovania asociačných pravidiel je generovanie frekventovaných množín v každom kroku výpočtu, pre každý vytvorený model Oracle umožňuje zobrazit' frekventované množiny. Výsledkom je zoznam množín s najväčšou podporou, doplnené o ID množiny, názov prvku a počet prvkov v množine. Celkový počet množín je voliteľný.

Asociačné pravidlá:

Cieľom tretej stránky je umožniť užívateľovi prehľad vydolovaných asociačných pravidiel z vytvoreného modelu. Asociačné pravidlá je možné podľa voľby filtrovať.

4.3.3 Prepojenie

Moje prvé kroky zoznamovania s podporou Oracle Data Mining prebehli na školskom serveri na verzii Oracle 10gR2. Zakladanie tabuliek a vytváranie modelov prebiehalo pomocou konzoly SQL*Plus. Po získaní prístupu k databáze a vývojovému prostrediu Application Express na Oracle 11g bolo potrebné založiť totožné databázové objekty ako na nižšej verzii. Oracle 11g nie je aktuálne na FITe sprístupnený pre účely študentov. Vzhľadom na jeho „asi“ nie úplné sprístupnenie alebo skôr kvôli nepostačujúcim oprávneniam, ktoré mi boli priradené, sa mi nepodarilo využívať jeho podporu pre dolovanie dát. Tu nastal problém ako prepojiť grafické rozhranie a programové balíky medzi dvomi databázami. Tento problém sa vyriešil pomocou databázového linku, ktorý po vytvorení umožnil v databáze Oracle 11g pristupovať k objektom v druhej databáze.

Použité databázové prepojenie:

```
CREATE DATABASE LINK "DBBERTA"  
  TO "DMUSER8" IDENTIFIED BY VALUES 'mypassword'  
  USING '(DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP) (HOST = berta.fit.vutbr.cz) (PORT = 1521))  
    ) (CONNECT_DATA =
```

```
(SERVICE_NAME = stud))) '
```

Jeho následné využitie pri volaní objektov:

```
begin
    ODM_asociations.create_model@dbberta('TAB_NAME', 'COLL_NAME' );
end;
```

Ak sa v databáze nachádza viacero schém s totožnými objektmi, je pri ich volaní nutné rozlišovať vlastníka. To by nastalo, ak by sme pre dolovanie používali dáta z oboch databáz. V aplikácii takúto identifikáciu nebolo potrebné použiť aj preto, že schéma, v ktorej sa objekt nachádza, je definovaná už v databázovom linku.

Objekt by sa volal nasledovne:

```
begin
    DMUSER8.ODM_asociations.create_model@dbberta('TAB_NAME', 'COLL_NAME' );
end;
```

Takéto použitie databázových linkov a definovanie vlastníkov je neefektívne pre nasledovné inštalovanie aplikácie a balíkov. Pri rozsiahlejších aplikáciách, kde je dopredu stanovené používanie linkov alebo nutnosť rozlišovania schém, sa táto funkčnosť rieši už pri inštalácii. Iné riešenie plynie pri použití synonym. Pri vytvorení synonyma by sa použil celý odkaz na objekt "owner.pacgage_name.function_name@dblink" a pri volaní by sa používal iba všeobecný príkaz "pacgage_name.function_name". Tento spôsob prepojenia databáz spraví aplikáciu jednoducho širiteľnú, pretože pri jej programovaní sa použijú synonymá a riešenie problémov ohľadom zdroju dát sa vyrieši jednoduchým spusteným skriptu na založenie synonym. Pre tvorenie synonym je ale dôležitá spolupráca s administrátorom databáz, pretože prístup do schém pomocou synonym závisí na pridelených právach užívateľa.

Vytvorenie synonyma:

```
create PUBLIC synonym "ODM_asociations.edit_view"
for "ODM_asociations.edit_view"@dbberta
/
```

4.4 Ukážka aplikácie

V tejto podkapitole bude popísaná krátky postup obsluhy aplikácie.

Krok č. 1: Nastavenie modelu

Pri vstupe do aplikácie sa zobrazí úvodná obrazovka nastavenia modelu. Pred dolovaním je vhodné mať predstavu o dátach nad ktorými idete dolovať, aby ste vedeli aké je optimálne nastavenie. Pri

analýze nákupného košíku sú dáta veľmi riedke, to znamená, že jednej transakcii je len malé percento produktov zakúpených z celého počtu ponúkaných. Z toho plynie aj nízka hodnota prahovej hranice. Pri jednom miliónе záznamov nákupných transakcií je prahová hranica podpory 1% celkom vysoká hodnota. Očakávané frekventované množiny v tejto situácii by boli obsiahnuté minimálne v 10 000 transakcií. Pre testovanie som nemal k dispozícii takto rozsiahle dátové sety a pre nastavenie stačí rozhranie podpory od 1-40%. Pri nastavení je vidieť zobrazenie aktuálnych hodnôt(viz obr. 4.1), ktoré ostali v aplikácii po predošlom dolovaní.

Oracle Data mining

Nastavenia modelu

<u>Podpora min.hodnota[%]</u>		Akt. hodnota 20	
<u>Spôľahlivosť min. podpora[%]</u>		Akt. hodnota 20	
<u>Max dĺžka pravidiel</u>		Akt. hodnota 3	Nové nastavenia
<u>Zvoľte tabuľku</u>	no data selected		Vytvoriť
<u>Zvoľte tabuľku prvkov</u>	no data selected		

Obr.4.1: Úvodná strana aplikácie

Po zvolení podpory testovacích hodnôt stačí stlačiť tlačidlo VYTVORIŤ a počkať kým prebehne vytvorenie modelu(viz obr. 4.2). V tomto prípade generovanie trvá do 30 sekúnd. Pre získavanie užitočných znalostí je vhodné vyskúšať viacero nastavení a zanalyzovať vygenerované pravidlá.

Oracle Data mining

Nastavenia modelu

Podpora min.hodnota[%]
10
Akt. hodnota 20

Spoľahlivosť min. podpora[%]
15
Akt. hodnota 20

Max dĺžka pravidla
Akt. hodnota 3

Zvoľte tabuľku
RESTAURANT_NESTED

Zvoľte stĺpec
REST_ID

Zvoľte tabuľku prvkov
REST_ID
REST_NAME
COFFEE_ID
COFFE_NAME

Item Id

Item Name

Nové nastavenia

Vytvoriť

Obr.4.2: Vytvorenie modelu

Krok č.2: Prehľad frekventovaných množín. Prehľad zobrazuje zvolený počet množín, usporiadaných podľa podpory. Počet zobrazených prvkov sa môže líšiť vzhľadom na početnosť množín. Prvky z jednej množiny sú zoradené po sebe. Pri vytváraní modelu s vysokými hodnotami podpory a spoľahlivosti nemusíme objaviť viacprvkové frekventované množiny. Tie majú väčšinou oveľa nižšiu podporu ako jednoprvkové množiny. Pre nájdenie viac prvkových množín je potrebné otestovať nastavenia modelu a následne zobraziť hľadané množiny.

Frekventované množiny			
Počet množín	10	Zobraziť	
Číslo množiny	Názov prvku	Počet prvkov v množine	Podpora
70	Excellent Service	1	0.52132
81	Excellent Food	1	0.47314
33	Excellent Decor	1	0.34209
83	Excellent Food	2	0.34209
83	Excellent Service	2	0.34209
59	Good Decor	1	0.30282
16	Weekend Dining	1	0.29800
52	Private Rooms Available	1	0.28282
39	Wheelchair Access	1	0.28066
80	Extraordinary Food	1	0.27078
45	Excellent Decor	2	0.26596
45	Excellent Service	2	0.26596

1 - 12

Obr.4.3: Výpis frekventovaných množín

Krok č.3: Výpis a práca s asociačnými pravidlami. Na záložke asociačné pravidlá je vytvorený formulár na prehľad získaných asociačných pravidiel. Užívateľ má možnosť podľa vlastného rozhodnutia a záujmu pracovať s pravidlami. Výsledky asociačnej analýzy bývajú prevažne plné zbytočných a už známych vzorov. Na to slúžia užívateľovi jednotlivé filtrovacie možnosti. Tie užitočné znalosti treba medzi tým množstvom zbytočných objaviť. Niekedy výsledky analýzy môžu byť skreslené aj nevhodnými dátami. Keď ich je malé množstvo, alebo keď nie sú vhodne predspracované. Pri dátach nákupného košíka sú výsledkom boolovské jednodimenzionálne pravidlá. Pri nich nás zaujíma iba či sa v košíku vyskytujú alebo nie, preto u formátu pravidiel môžeme odmyslieť hodnoty 1, ktoré sú súčasťou pravidla (napr. Excellent Decor= 1) . Na prvý pohľad sa môže zdať, že sme získali pravidlá obsahujú jeden , maximálne dva prvky. Dôsledkom chýbania dlhších pravidiel je nastavenie minimálnej hranice podpory, ktorú nesplňajú. Po zmene nastavenia , je možné nad týmito dátami získať pravidlá aj s väčšou dĺžkou. Takto získane pravidlá sa v reálnom svete podstúpia na analýzu odborníkovi na danú problematiku a vzorku dát, ktorý dokáže zhodnotiť získane pravidlá a ich vypovedaciu hodnotu pre použitie v praxi.

Asociačné pravidlá				
Počet zobrazených pravidiel	10	Zobraziť		
Min. podpora		Min. dĺžka pravidla		
Min. spoľahlivosť		Max. dĺžka pravidla		
Pravidlo	Podpora	Spoľahlivosť	Antecedent	Consequent
35	0.17220	0.89940	Excellent Decor= 1, Excellent Food= 1,	Excellent Service= 1
42	0.13010	0.90450	Private Parties= 1, Excellent Service= 1,	Private Rooms Available= 1
32	0.11710	0.98380	Private Rooms Available= 1, Weekend Dining= 1,	Private Parties= 1
31	0.11710	0.91010	Private Parties= 1, Weekend Dining= 1,	Private Rooms Available= 1
14	0.11470	0.99580	Private Rooms Available= 1, Parking/Valet= 1,	Private Parties= 1
12	0.11470	0.98760	Private Parties= 1, Parking/Valet= 1,	Private Rooms Available= 1
24	0.10700	0.91360	Private Rooms Available= 1, Wheelchair Access= 1,	Private Parties= 1
104	0.10430	1.00000	\$15-\$30= 1,	Good Food= 1
105	0.10430	0.99770	Good Food= 1,	\$15-\$30= 1
75	0.10140	0.90930	Private Parties= 1, Excellent Food= 1,	Private Rooms Available= 1

1 - 10

Obr.4.4: Výpis asociačných pravidiel bez filtrovania

Nastavenie modelu

Asociačné pravidlá

Počet zobrazených pravidiel Zobraziť

Min. podpora Min. dĺžka pravidla

Min. spoľahlivosť Max. dĺžka pravidla

Pravidlo	Podpora ▼	Spoľahlivosť	Antecedent	Consequent
35	0.17220	0.89940	Excellent Decor= 1, Excellent Food= 1,	Excellent Service= 1
36	0.17220	0.64760	Excellent Service= 1, Excellent Decor= 1,	Excellent Food= 1
34	0.17220	0.50350	Excellent Service= 1, Excellent Food= 1,	Excellent Decor= 1

1 - 3

Obr.4.5: Výpis asociačných pravidiel s filtrovaním

5 Záver

5.1 Zhodnotenie dosiahnutých výsledkov

Účelom tejto práce bolo zoznámiť sa s problematikou získavania znalostí, prípravou dát a podporu DB serveru Oracle pre Data Mining a výsledkom je ukážková aplikácia, v ktorej boli získané poznatky použité.

Prevažnú väčšinu času som strávil nad študovaním podpory Oraclu, jeho podporných balíkov a ich následné využitie pre vývoj aplikácie. Výsledkom bolo pochopenie funkcionality získavania znalostí pomocou PL/SQL v Oracli.

Už menším problémom bol dátový typ `NESTED_TABLE`, s ktorým som sa doposiaľ nestretol. Tento dátový typ nepodporovali niektoré konzoly a inštancie Oraclu. Tento dátový typ sa používal pre formátovanie relačných tabuliek na transakčné, ktoré Oracle podporuje ako jediný formát vstupných dát pre dolovanie asociačných pravidiel.

Táto práca popisuje postup od prípravy dát, nastavovania a samotného dolovania asociačných pravidiel a následne ich prezentácia pomocou grafického rozhrania. Popisuje problémy, ktoré sa vyskytli pri práci s PL/SQL dotazmi a ich riešenia. Popisuje riešenie prepojenia databáz a využívanie synonym ako riešenie obecných dotazov na databázovými objektmi.

Za hlavný úspech tejto práce považujem vytvorenie ukážkovej aplikácie, ktorá bola navrhnutá tak, aby bola jednoducho prístupná cez webový prehliadač a jednoducho širiteľná. Tento cieľ bol dosiahnutý použitím dynamického PL/SQL a generovania pohľadov podľa voľby užívateľa.

Výsledná aplikácia spĺňa požadované vlastnosti a je vytvorená tak, aby oboznámený užívateľ bez problémov dokázal úspešne získavať asociačné pravidlá a frekventované množiny nad zvolenými dátami. Jednoduchým ovládaním je zabezpečené užívateľsky prívetivé prostredie na dolovanie asociačných pravidiel. Možno nezdarom sú problematické sady dát, ktoré okrem gramatických chýb majú malú vypovedaciu hodnotu. Táto aplikácia slúži na získavanie frekventovaných množín a asociačných pravidiel, ale závisí len na užívateľovi a jeho vedomostiach v oblasti získavania znalostí, aby zo získaných dát dokázal získať užitočné a netriviálne znalosti.

5.2 Diskusia o možnostiach rozšírenia aplikácie

Keďže na fakulte FIT prebieha vývoj aplikácie na dolovanie využívajúc Oracle podporu pre Javu a PL/SQL podpora bola využitá na vytvorenie aplikácie prvýkrát v tejto práci. Keďže Oracle Data Mining podpora je pomerne rozsiahla, tak usudžujem, že poskytuje dostatok možností pre budúce rozšírenie aplikácie.

Ako prvé by som navrhoval implementovať export a import dolovacích modulov. Táto funkčnosť by umožnila využívanie vytvorených modulov aj v iných aplikáciách, ktoré sú založené na architektúre dolovania pomocou Oracle Data Mining.

Ďalší možný prístup k rozšíreniu aplikácie je zamerať sa na import a transformáciu dát, aby nebolo potrebné na vkladanie dát používať konzolu.

Obsahom tejto práce bolo dolovanie asociačných pravidiel, takže ďalšie rozšírenie by sa mohlo zamerať na implementáciu ďalších Oraclom podporovaných dolovacích úloh.

Literatúra

- [1] Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Second Edition. Elsevier Inc., 2006, 770 p., ISBN 1-55860-901-3.
- [2] Zendulka J., Bartík V., Lukáš R., Rudolflová I.: Získávání znalostí z databází, studijní opora, Brno, 2006, FIT VUT v Brně.
- [3] Berka Petr : Dobývání znalostí z databází, ACADEMIA, 2003, 366.p , ISBN 80-200-1062-9.
- [4] Oracle® Database, PL/SQL Packages and Types Reference 11g Release 1 (11.1.) April 2008
Dokument dostupný na URL:
http://download-uk.oracle.com/docs/cd/B28359_01/appdev.111/b28419/toc.htm (24.4.2009)
- [5] Oracle® Database, Administrator's Guide, 11g Release 1 (11.1), B28130-04, August 2008
Dokument dostupný na URL:
http://download-uk.oracle.com/docs/cd/B28359_01/appdev.111/b28419/toc.htm (5.4.2009)
- [6] Oracle® Database, Application Developer's Guide, 11g Release 1 (11.1), B28131-04, May 2008
Dokument dostupný na URL:
http://download.oracle.com/docs/cd/B28359_01/datamine.111/b28131/title.htm (22.4.2009)

Zdrojové dáta pre dolovanie

Pre testovacie účely aplikácie boli použité dáta z predmetu Získavanie znalostí z databáz na VUT FIT.

Zoznam príloh

Príloha 1. Zdrojové texty sú obsiahnuté na CD/DVD aj so súborom README, kde je popis práce s aplikáciou

Príloha 2. CD/DVD