Data Mining i odkrywanie wiedzy w bazach danych

Maciej Zakrzewicz
Instytut Informatyki
Politechniki Poznańskiej

mzakrz@cs.put.poznan.pl

Streszczenie

Intensywnie rozwijająca się dziedzina odkrywania wiedzy w bazach danych (*Knowledge Discovery in Databases*) oraz eksploracji danych (*Data Mining*) jest odpowiedzią na gwałtowny wzrost ilości informacji gromadzonych w bazach i magazynach danych. Techniki eksploracji danych pozwalają na znajdowanie wcześniej nieznanych zależności i schematów, które mogą być wykorzystane do wspomagania podejmowania decyzji lub opisu bazy danych. Artykuł prezentuje podstawową problematykę związaną z odkrywaniem wiedzy w bazach danych. Omówione są techniki eksploracji danych, metody reprezentacji odkrywanej wiedzy oraz jej praktyczne zastosowania. Przedstawiono również wybrane produkty komercyjne w kontekście realizacji zadań procesu odkrywania wiedzy.

1. Wprowadzenie

Postęp technologiczny w zakresie cyfrowego generowania i gromadzenia informacji doprowadził do przekształcenia się baz danych wielu przedsiębiorstw, urzędów i placówek badawczych w zbiorniki ogromnych ilości danych. Na bezprecedensowy, wielki rozrost systemów bazodanowych złożyło się przede wszystkim upowszechnienie kodów paskowych i kart płatniczych oraz pojawienie się szybszych, pojemniejszych i tańszych pamięci masowych. Dla przykładu, baza danych wykorzystywana przez sieć sprzedaży *Wal-Mart* gromadzi dziennie informacje o ponad 20 milionach transakcji [[3]]. Przedsiębiorstwo *Mobil Oil* rozwija magazyn danych pozwalający na przechowywanie ponad 100 terabajtów danych związanych z wydobyciem ropy naftowej [[3]]. System satelitarnej obserwacji *EOS* zbudowany przez *NASA* generuje w ciągu każdej godziny dziesiątki gigabajtów danych obrazowych [[3]]. Nawet niewielkie supermarkety rejestrują codziennie sprzedaż tysięcy artykułów. Nasze możliwości analizowania i rozumienia tak dużych wolumenów danych są dużo mniejsze od możliwości ich zbierania i przechowywania.

Zebrane w bazach danych zapisy o np. dotychczasowej działalności przedsiębiorstwa, poziomie i strukturze sprzedaży oraz cechach klientów mogą być wykorzystane do wspomagania podejmowania decyzji o dalszym kształtowaniu sprzedaży i kierunkach marketingu przedsiębiorstwa. Komputerowe systemy wspomagania decyzji (*Decision Support Systems*) bazują na zgromadzonej wiedzy ekspertów, po części pochodzącej z analizy zawartość baz danych. Na prostą analizę baz danych pozwalają środowiska typu *OLAP* (*Online Analytical Processing*), które umożliwiają wielowymiarową obserwację agregowanych wartości wybranych atrybutów jednej lub wielu połączonych

relacji. Metodologia *OLAP* zakłada, że użytkownik przygotowuje pewną hipotezę, której poprawność weryfikuje korzystając z narzędzi *OLAP* (np. *Oracle Express Server*). Przykładowo, ekspert może podejrzewać, że sprzedaż obuwia letniego jest w jakiś sposób uzależniona od lokalizacji sklepu i od miesiąca w roku. Aby przeprowadzić analizę takiej hipotezy, ekspert może przy pomocy narzędzi *OLAP* wyznaczyć sumę wartości sprzedaży z poprzednich lat w odniesieniu do różnych rejonów kraju i różnych miesięcy. Obserwacja wyników (również uzupełniona prezentacją graficzną) pozwoli zauważyć, że największa sprzedaż obuwia letniego występuje w miejscowościach nadmorskich w lipcu i sierpniu. Ograniczenia takiej metody związane są z koniecznością przygotowywania hipotez, które podlegają późniejszej weryfikacji. W ten sposób jakość wiedzy wykorzystywanej przez systemy wspomagania decyzji ograniczona jest kreatywnością i wyobraźnią eksperta. Istnieje także niebezpieczeństwo akceptacji hipotez fałszywych.

Pozbawione wymienionych wad systemów *OLAP* jest automatyczne znajdowanie wiedzy, będące przedmiotem dynamicznie rozwijającej się dziedziny **odkrywania wiedzy w bazach danych** (*Knowledge Discovery in Databases*) i jej technologii **eksploracji danych** (*Data Mining*). Odkrywanie wiedzy nie wymaga przygotowywania hipotez przez ekspertów - są one automatycznie generowane i automatycznie weryfikowane. Zadania ekspertów sprowadzają się do oceny i akceptacji odkrytej wiedzy, najczęściej poprzez kontrolowanie jej wskaźników statystycznych. Dla przykładu, w procesie odkrywania wiedzy ekspert wskazuje zbiór danych o sprzedaży obuwia i ustala, że zależności interesujące to te, które są spełnione przez conajmniej 60% transakcji sprzedaży. W odpowiedzi ekspert uzyskuje zbiór wszystkich zależności, jakie zachodzą w conajmniej 60% bazy danych o sprzedaży obuwia. Znalezione zależności są zwykle bardziej precyzyjne niż pojawiające się w procesie *OLAP*, np. zależność "obuwie letnie jest najczęściej kupowane w lipcu, w miejscowościach województwa gdańskiego, przez kobiety w wieku 18-25 lat".

Istnieje wiele zastosowań odkrywania wiedzy w bazach danych. We wspomnianej powyżej problematyce wspomagania decyzji podstawowym celem odkrywania wiedzy jest automatyzacja **budowy baz wiedzy**. Zgromadzona wiedza umożliwia np. przewidywanie nieznanych wartości wybranych atrybutów relacji na podstawie zadanych wartości pozostałych atrybutów. Innym zastosowaniem odkryć wiedzy jest inteligentne i zautomatyzowane **konstruowanie opisu bazy danych**. Wynikiem takiego działania jest specyfikacja charakterystyki bazy danych, która może być wykorzystana np. do znalezienia nowych zależności funkcyjnych. Wreszcie odkrywanie wiedzy może pozwolić na znajdowanie tzw. **anomalii w danych**, czyli tych krotek w relacji, których charaktrystyka odbiega od statystycznie dominującej charakterystyki całego zbioru danych. Powszechnym wykorzystaniem wyszukiwania anomalii jest automatyczne wykrywanie oszustw podatkowych i ubezpieczeniowych.

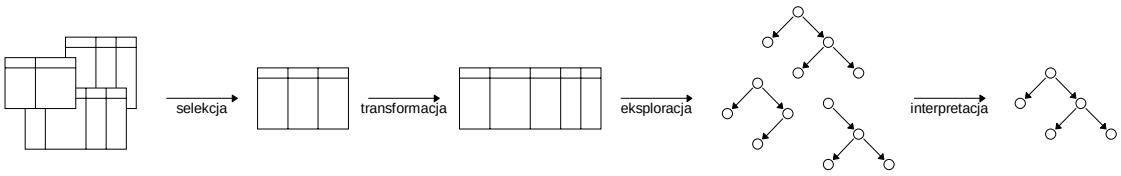
Odkrywanie wiedzy jest postrzegane jako złożony proces selekcji i transformacji danych, ich eksploracji, a następnie interpretacji uzyskanych wyników. Wszystkie te operacje muszą być wspierane przez specjalizowane oprogramowanie, nazywane **systemem odkrywania wiedzy** (*Knowledge Discovery in Databases Management System*), współpracujące z systemem zarządzania bazą danych (*DBMS*) i udostępniające interfejs *API* w architekturze klient-serwer. Na rynku dostępnych jest kilka pakietów programowych oferujących różne techniki eksploracji danych zgromadzonych w popularnych systemach bazodanowych (*Oracle, Informix, Sybase, Ingres*), wraz ze szczegółowym przygotowaniem danych i graficzną formą prezentacji wyników.

2. Odkrywanie wiedzy

Odkrywanie wiedzy w bazach danych polega na wyszukiwaniu czytelnych schematów i wzorców, które nie były wcześniej znane, a są potencjalnie użyteczne dla wspomagania decyzji i charakteryzowania bazy danych. Odkrywanie wiedzy korzysta z wielu doświadczeń i metod dziedzin sztucznej inteligencji i uczenia maszynowego. Główne problemy odkrywania wiedzy w bazach danych wiążą się z koniecznością przetwarzania bardzo dużych wolumenów danych oraz potrzebą interakcyjnego wyszukiwania wiedzy przez wielu współbieżnie pracujących użytkowników.

Odkrywanie wiedzy jest procesem złożonym, którego realizacja polega na przygotowaniu danych, ich eksploracji oraz interpretacji wyników. Najczęściej proces odkrywania wiedzy w bazach danych składa się z kolejnych kroków (א):

1. **Selekcja danych** - wybór relacji i krotek, które będą eksplorowane, definicja sposobu łączenia relacji,
2. **Transformacja danych** - konwersja typów atrybutów, definicja atrybutów wywiedzionych, dyskretyzacja wartości ciągłych,
3. **Eksploracja** - ekstrakcja wiedzy z danych: generowanie reguł, drzew decyzyjnych, sieci neuronowych itp.
4. **Interpretacja wyników** - wybór najbardziej interesującej wiedzy, logiczna i graficzna wizualizacja wyników,



Rys. 1 Fazy odkrywania wiedzy

Kluczową fazą procesu odkrywania wiedzy jest eksploracja danych (*Data Mining*). Celem eksploracji jest wykorzystanie właściwego algorytmu dla znajdowania zależności i schematów w przygotowanym zbiorze danych, a następnie ich reprezentacja w postaci formalnej, zrozumiałej dla użytkownika. Najpopularniejszymi formami reprezentacji odkrywanej wiedzy są drzewa decyzyjne i reguły logiczne. Warto odnotować, że w prasie fachowej i materiałach reklamowych często zamiennie stosuje się terminy: eksploracja danych (*Data Mining*), odkrywanie wiedzy w bazach danych (*Knowledge Discovery in Databases*) i eksploracja baz danych (*Database Mining*).

3. Techniki eksploracji danych

Eksploracja danych posługuje się różnymi technikami, które budują specyficzne rodzaje wiedzy. W zależności od przeznaczenia odkrywanej wiedzy, może ona odwzorowywać klasyfikacje, regresje, klastrowanie, charakterystyki, dyskryminacje, asocjacje itp.. Poniżej dokonano krótkiej charakterystyki każdej z wymienionych technik eksploracji danych.

3.1 Klasyfikacja

Klasyfikacja polega na znajdowaniu sposobu odwzorowania danych w zbiór predefiniowanych klas. Na podstawie zawartości bazy danych budowany jest model (np. drzewo decyzyjne, reguły logiczne), który służy do klasyfikowania nowych obiektów w bazie danych lub głębszego zrozumienia istniejących klas. Przykładowo, w medycznej bazie danych znalezione mogą być reguły klasyfikujące poszczególne schorzenia, a następnie przy pomocy znalezionych reguł automatycznie może być przeprowadzone diagnozowanie kolejnych pacjentów. Inne przykłady zastosowań klasyfikacji to:

- rozpoznawanie trendów an rynkach finansowych,
- automatyczne rozpoznawanie obiektów w dużych bazach danych obrazów,
- wspomaganie decyzji przyznawania kredytów bankowych.

3.2 Regresja

Regresja jest techniką polegającą na znajdowaniu sposobu odwzorowania danych w rzeczywistoliczbowe wartości zmiennych predykcyjnych. Przykłady zastosowań regresji:

- przewidywanie zawartości biomasy obecnej w ściółce leśnej na podstawie dokonanych zdalnych pomiarów mikrofalowych
- szacowanie prawdopodobieństwa wyzdrowienia pacjenta na podstawie przeprowadzonych testów diagnostycznych

3.3 Klastrowanie

Klastrowanie (clustering) polega na znajdowaniu skończonego zbioru kategorii opisujących dane. Kategorie mogą być rozłączne, zupełne, mogą też tworzyć struktury hierarchiczne i nakładające się. Przykładowo, zbiór danych o nieznanych chorobach może zostać w wyniku klastrowania podzielony na szereg grup cechujących sie najsilniejszym podobieństwem symptomów. Innymi przykładami zastosowań klastrowania mogą być:

- określanie segmentów rynku dla produktu na podstawie informacji o klientach
- znajdowanie kategorii widmowych spektrum promieniowania nieba

3.4 Odkrywanie charakterystyk

Odkrywanie charakterystyk (summarization, characterization) polega na znajdowaniu zwięzłych opisów (charakterystyk) podanego zbioru danych. Przykładowo, symptomy określonej choroby mogą być charakteryzowane przez zbiór reguł charakteryzujących. Inne przykłady odkrywania charakterystyk to:

- znajdowanie zależności funkcyjnych pomiędzy zmiennymi
- określanie powszechnych symptomów wskazanej choroby

3.5 Dyskryminacja

Dyskryminacja polega na znajdowaniu cech, które odróżniają wskazaną klasę obiektów (*target class*) od innych klas (*contrasting classes*). Przykładowo, zbiór reguł dyskryminujących może opisywać te cechy objawowe, które odróżniają daną chorobę od innych.

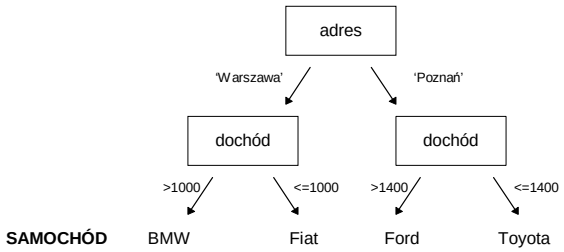
3.6 Odkrywanie asocjacji

Odkrywanie asocjacji (association discovery) polega na znajdowaniu związków pomiędzy występowaniem grup elementów w zadanych zbiorach danych. Najpopularniejszym przykładem odkrywania asocjacji jest tzw. analiza koszyka - przetwarzanie baz danych supermarketów i hurtowni w celu znalezienia grup towarów, które są najczęściej kupowane wspólnie. Przykładowo, znalezione asocjacje mogą wskazywać, że kiedy klient kupuje słone paluszki, wtedy kupuje także napoje gazowane.

4. Reprezentacja wiedzy

Wiedza odkrywana przy wykorzystaniu technik eksploracji danych może być reprezentowana i przechowywana w różnych formach. W dziedzinach uczenia maszynowego i sztucznej inteligencji dla potrzeb przechowywania wiedzy stosowane są struktury takie jak: sieci neuronowe, drzewa decyzyjne, listy decyzyjne, sieci semantyczne, proste i złożone reguły logiczne. Nie wszystkie z tych struktur spełniają wymagania narzucane przez problematykę odkrywania wiedzy w bazach danych. Podstawowym wymogiem jest prostota opisu i czytelność reprezentowanej wiedzy dla użytkownika. Najpopularniejszymi i najbardziej praktycznymi metodami reprezentacji wiedzy odkrywanej w bazach danych są drzewa decyzyjne i reguły logiczne.

Adres	**Dochód**	**Samochód**
Warszawa	4000	BMW
Poznań	2900	Ford
Poznań	1400	Toyota
Warszawa	1000	Fiat
Poznań	1600	Ford
Poznań	3500	Ford



Rys. 2 Drzewo decyzyjne

4.1 Drzewa decyzyjne

Modele drzew decyzyjnych są najpowszechniejszą formą reprezentacji wiedzy odkrywanej w wyniku eksploracji danych przez dostępne dziś oprogramowanie komercyjne. Drzewo decyzyjne jest formą opisu wiedzy klasyfikującej. Węzły drzewa są opisane przez atrybuty eksplorowanej relacji. Krawędzie drzewa określają możliwe wartości dla

atrybutu. Liśćmi drzewa są wartości atrybutu klasyfikującego. Klasyfikacja odbywa się poprzez przeglądanie drzewa od korzenia do liści przez krawędzie opisane wartościami atrybutów. Przykład drzewa decyzyjnego dla prostej relacji przedstawiony został na ב. Atrybutem klasyfikującym jest *Samochód*. Z przedstawionego przykładowego drzewa decyzyjnego można odczytać, że np. mieszkańcy Poznania, o dochodach nie przekraczających 1400 złotych kupują najczęściej samochody marki Toyota.

4.2 Reguły logiczne

Poważną wadą drzew decyzyjnych jest ich tendencja do przybierania w praktyce bardzo dużych rozmiarów, co powoduje trudności w rozumieniu i analizowaniu drzew przez użytkowników. Bardziej efektywna i silniejsza reprezentacja wiedzy jest możliwa przy użyciu reguł logicznych. Reguły logiczne są formułami zapisanymi w postaci implikacji typu:

$$r_1(a_1, v_1) \wedge r_2(a_2, v_2) \wedge \ldots \wedge r_j(a_j, v_j) \rightarrow r_k(a_k, v_k) \wedge r_l(a_l, v_l) \wedge \ldots \wedge r_n(a_n, v_n)$$

gdzie:

- a_i jest atrybutem,
- v_i jest wartością prostą (np. liczba, ciąg znaków) lub złożoną (np. zbiór),
- r_i jest predykatem (np. równość, zawieranie)

Lewą stronę reguły nazywa się **ciałem reguły** (body), jej prawą stronę - **głową reguły** (head). Definiuje się dwie relacje, jakie mogą zachodzić pomiędzy danymi a regułami: **potwierdzanie** i **naruszanie**. Krotka (lub krotki) **potwierdza** regułę, jeżeli dla wartości jej atrybutów zarówno ciało, jak i głowa reguły przyjmują wartości logicznej prawdy. Krotka (lub krotki) **naruszają** regułę, jeżeli dla wartości jej atrybutów ciało reguły przyjmuje wartość logicznej prawdy, a głowa reguły - wartość logicznego fałszu. Każda reguła posiada dodatkowo dwa wskaźniki statystycznej ważności i siły: **wsparcie** (support) i **zaufanie** (confidence). Wsparciem reguły jest liczba (lub procent) krotek relacji, które potwierdzają regułę. Zaufanie reguły wyraża się zależnością:

$$c(rule) = s(rule) / s(body(rule))$$

gdzie:

- $c(rule)$ jest wartością zaufania dla reguły
- $s(rule)$ jest wartością wsparcia dla reguły
- $s(body(rule))$ jest wartością wsparcia dla ciała reguły, czyli liczbą (lub procentem) krotek relacji, dla których ciało reguły przyjmuje wartość logicznej prawdy

W zależności od zastosowanej techniki eksploracji danych, otrzymane reguły nazywa się klasyfikującymi, asocjacyjnymi, dyskryminującymi itd. Przykładowa relacja oraz znalezione w niej reguły klasyfikujące (*classification*

rules) według atrybutu *Diagnoza* zostały przedstawione na ג. Symbol *S* oznacza wsparcie reguły, symbol *C* oznacza jej zaufanie.

Temperatura	Ból_głowy	Ból_gardła	Diagnoza
wysoka	tak	nie	zatrucie
wysoka	tak	nie	zdrowy
wysoka	tak	tak	angina
wysoka	nie	tak	angina

```
Ból_gardła = „tak” → Diagnoza = „angina” (S=50% C=100%)
Temperatura=”wysoka” ∧ Ból_głowy=”tak” ∧ Ból_gardła=”nie” → Diagnoza=”zatrucie” (S=25% C=100%)
Temperatura=”wysoka” ∧ Ból_głowy=”tak” ∧ Ból_gardła=”nie” → Diagnoza=”zdrowy” (S=25% C=100%)
```

Rys. 3 Reguły klasyfikujące

Pierwsza z powyższych reguł stwierdza, że jeżeli u pacjenta występuje ból gardła, to jest on ze 100% prawdopodobieństwem chory na anginę, a schemat taki został zaobserwowany w połowie wszystkich rekordów relacji.

Bardzo interesującą klasą reguł logicznych są reguły reprezentujące asocjacje odkryte w bazie danych. Takie reguły są nazywane regułami asocjacyjnymi (association rules). Na ד przedstawiono przykład relacji, w której rejestrowana jest sprzedaż w supermarkecie, oraz zbiór najsilniejszych reguł asocjacyjnych odkrytych w tej relacji. Asocjacje są znajdowane w grupach towarów zakupionych w ramach jednej transakcji (wspólna wartość atrybutu *Trans_id*). Symbol *S* oznacza wsparcie reguły, symbol *C* oznacza jej zaufanie.

Trans_id	Klient	Produkt
1	101	Chleb
1	101	Mleko
2	100	Masło
2	100	Chleb
2	100	Mleko
3	105	Masło
4	100	Wino
4	100	Mleko
4	100	Gazeta

```
Produkt=”Mleko” → Produkt=”Chleb” (S=50% C=66%)
Produkt=”Chleb” ∧ Produkt=”Mleko” → Produkt=”Masło”  (S=25% C=50%)
Produkt=”Masło” ∧ Produkt=”Mleko” → Produkt=”Chleb”  (S=25% C=100%)
Produkt=”Mleko” → Produkt=”Masło” ∧ Produkt=”Chleb”  (S=25% C=33%)
```

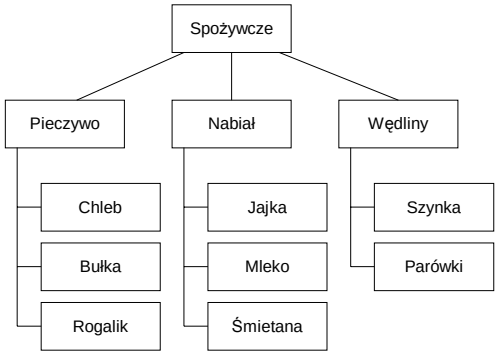
Rys. 4 Reguły asocjacyjne

Pierwsza z powyższych reguł stwierdza, że w dwóch przypadkach na trzy, klient który kupuje mleko, kupuje także chleb. Schemat taki występuje w połowie wszystkich transakcji.

5. Wiedza uzupełniająca

W swojej najprostszej postaci, proces eksploracji danych samodzielnie wyszukuje regularności i schematy w zbiorach danych. W wielu zastosowaniach dostępna jest jednak także taka wiedza dotycząca badanego obszaru, która nie została odkryta automatycznie lecz jest wynikiem doświadczenia ekspertów z danej dziedziny. Istnieją możliwości uwzględniania wiedzy dodatkowej przez algorytmy eksploracji danych. Uwzględnianie uzupełniających wskazówek, podpowiedzi, zależności i znanych schematów pozwala na automatyczne znajdowanie prostszych i silniejszych reguł i drzew decyzyjnych.

Wiedza eksperta, która uzupełnia proces eksploracji danych, jest najczęściej reprezentowana w postaci hierarchii generalizacji lub za pomocą atrybutów wyliczeniowych. Hirarchie generalizacji dostarczają informacji o prostej klasyfikacji wartości atrybutów w tzw. wartości uogólnione. Przykładem hierarchii generalizacji może być przedstawione na ה drzewo klasyfikujące sprzedawane w supermarkecie produkty w poszczególne kategorie. W przedstawionym drzewie wartości uogólnione dla wartości *„bułka”* to *„pieczywo”* i *„spożywcze”*. Hierarchia generalizacji jest definiowana dla jednego atrybutu relacji, jednak z każdym atrybutem może być związanych wiele takich hierarchii. Reguły logiczne odkrywane z wykorzystaniem hierarchii generalizacji nazywa się regułami uogólnionymi (*generalized rules*).



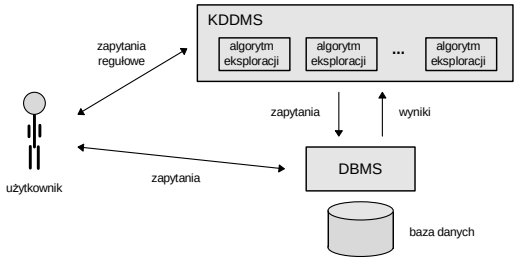
Rys. 5 Drzewo hierarchii generalizacji

Jeżeli wiedza dodatkowa dla procesu eksploracji posiada skomplikowaną strukturę, wtedy dla jej reprezentacji stosuje się atrybuty wyliczeniowe. Atrybut wyliczeniowy stanowi dodatkowy atrybut relacji, którego wartość nie jest przechowywana w relacji wraz z wartościami innych atrybutów, lecz jest każdorazowo wyliczana za pomocą zdefiniowanej przez użytkownika funkcji. Atrybuty wyliczeniowe są stosowane głównie do określanie metod

dyskretyzacji atrybutów o wartościach ciągłych. Atrybuty wyliczeniowe mogą również implementować drzewa hierarchii generalizacji.

6. Architektura systemu odkrywania wiedzy

Odkrywanie wiedzy w bazach danych musi być, ze względu na rozmiary rozwiązywanych problemów, szeroko wspomagane przez specjalizowane oprogramowanie, nazywane systemem odkrywania wiedzy (KDDMS) [[1]]. System odkrywania wiedzy jest zwykle integrowany z systemem zarządzania bazą danych (DBMS), co pozwala na szybki dostęp do eksplorowanych danych oraz umożliwia składowanie odkrytej wiedzy w bazie danych. W systemach, które odkrywają wiedzę w sposób interakcyjny, powszechną formą reprezentacji wiedzy są reguły logiczne. Na 1 przedstawiono ogólną architekturę systemu odkrywania wiedzy.



Rys. 6 Architektura systemu odkrywania wiedzy

Użytkownicy kierują do systemu odkrywania wiedzy zapytania, zwane zapytaniami regułowymi, w których specyfikują, jakich reguł poszukują oraz jakie dane mają być eksplorowane w celu odkrycia reguł. System odkrywania wiedzy wykorzystuje odpowiedni do żądanego typu reguł algorytm eksploracji danych. W celu znalezienia reguł, algorytm eksploracji wysyła zapytania do systemu zarządzania bazą danych. Znalezione reguły są następnie filtrowane tak, aby uwzględnić kryteria zapytania regułowego wystosowanego przez użytkownika. Na zakończenie, zbiór reguł zwracany jest użytkownikowi jako wynik jego zapytania.

6.1 Języki zapytań

Dla potrzeb komunikacji użytkowników i aplikacji z systemem odkrywania wiedzy zaproponowano szereg języków zapytań regułowych [[2],[4],[5],[9]]. Większość z rozwiązań proponuje rozszerzenia standardu języka SQL o dodatkowe operatory, polecenia i typy danych. Podstawową zaletą wynikającą z rozszerzania języka SQL jest wykorzystanie wspólnego interfejsu zarówno do wyszukiwania danych jak i do generowania reguł na podstawie tych danych. Przykładowo, język MineSQL [[9]] udostępnia mechanizmy językowe dla generowania reguł, ich gromadzenia, wyszukiwania i stosowania. Wprowadza nowy typ danych *RULE*, służący do przechowywania i operowania na regułach, oraz zbiór funkcji konwersji związanych z nowym typem danych. Pozwala konstruować zapytania znajdujące reguły asocjacyjne, klasyfikujące i dyskryminujące. Znalezione reguły mogą być przechowywane w relacjach bazy danych i wykorzystywane dla podejmowania decyzji i weryfikacji danych. Język MineSQL wspiera także definiowanie i stosowanie hierarchii generalizacji oraz atrybutów wyliczenowych. Zapytania regułowe języka MineSQL mogą być

także wykorzystywane jako podzapytania innych zapytań języka SQL. Na ז przedstawiono przykład zapytania regułowego wyrażonego w języku MineSQL, które generuje reguły przedstawione wcześniej na ד.

```
MINE rule, support(rule), confidence(rule)
FOR produkt
FROM sprzedaż
WHERE confidence(rule) >= 0.33
  AND support(rule) >= 0.25
GROUP BY trans_id
```

Rys. 7 Zapytanie regułowe w języku MineSQL

7. Praktyczne zastosowania

Dotychczasowe doświadczenia pokazują, że stosowanie systemów odkrywania wiedzy w bazach danych pozwala na znaczącą poprawę jakości produkcji oraz podniesienie poziomu zysków. Poniżej przedstawiono kilka najpopularniejszych „sukcesów" odkrywania wiedzy w bazach danych dużych przedsiębiorstw:

- **„Database Marketing"**

  „Database Marketing" polega na analizie danych o klientach w celu znajdowania schematów ich preferencji i następnie wykorzystywania tych schematów dla precyzyjnej selekcji kolejnych klientów. „Database Marketing" w *American Express* doprowadził do 10-15% wzrostu zakupów z wykorzystaniem kart kredytowych.

- **Weryfikacja poprawności danych**

  Reuters stosuje techniki eksploracji danych dla weryfikacji poprawności i wykrywania prawdopodobnych przekłamań w wysokości publikowanych kursów wymiany walut.

- **Profil klienta**

  BBC przy pomocy systemu eksploracji danych przewiduje profil widowni programów telewizyjnych w celu wyboru optymalnych pór ich nadawania.

- **Wykrywanie oszustw finansowych**

  Polega na znajdowaniu transakcji finansowych, których cechy odbiegają od statystycznie dominującej charakterystyki finansowej bazy danych.

8. Produkty komercyjne

Od niedawna na rynku dostępne są zintegrowane środowiska programowe, które umożliwiają odkrywanie wiedzy w najbardziej popularnych systemach zarządzania bazami danych. Poniżej przedstawione zostały cztery najbardziej zaawansowane produkty umożliwiające selekcję danych, ich różnorodną eksplorację oraz wizualizację i interpretację odkrytej wiedzy.

8.1 Intelligent Miner, IBM

Intelligent Miner [[6]] to zestaw narzędzi realizujących algorytmy odkrywania klasyfikacji i asocjacji, klastrowania, wykrywania odchyleń itp. Pozwala na eksplorację danych zgromadzonych w bazach DB2, Oracle lub *Sybase*, współpracując z *IBM DataJoiner* dla przygotowania danych. Jest zorientowany na realizację następujących zastosowań odkrywania wiedzy: segmentacja klientów, analiza koszyka i wykrywanie oszustw finansowych. *Intelligent Miner* pracuje m.in. w systemach *AIX*, *AS/400*, *OS/390*, korzystając z architektury klient-serwer.

8.2 MineSet, Silicon Graphics

Mine Set jest środowiskiem, które dostarcza narzędzi dla przygotowywania danych, eksploracji danych i wizualizacji wiedzy [[7]]. Wspierane metody eksploracji to: odkrywanie reguł asocjacyjnych, klasyfikacja za pomocą drzew decyzyjnych, klasyfikacja na podstawie niepełnych danych i szacowanie klasyfikującej siły atrybutów relacji. Ponadto, *Mine Set* umożliwia animację i trójwymiarową wizualizację danych, drzew decyzyjnych i reguł. Środowisko pracuje na komputerach *SGI O2*, *Octane*, *Onyx*, *Origin 200*, *Origin 2000*, *Indy*, *Indigo2*, *Onyx* i *Challenge*. Dane mogą być pobierane bezpośrednio z systemów baz danych *Oracle*, *Informix* i *Sybase*.

8.3 Clementine, Integral Solutions

Pakiet umożliwiający znajdowanie klasyfikacji w danych pobieranych z baz typu *Oracle*, *Ingres*, *Sybase* i *Informix*, z plików tekstowych lub z arkuszy kalkulacyjnych [[10]]. Możliwa jest szeroka selekcja danych, łączenie krotek, definiowanie atrybutów wywiedzionych. Dane mogą być przedstawiane w postaci graficznej. System wykorzystuje sieci neuronowe, drzewa decyzyjne i reguły. Jest wyposażony w interfejs programowania graficznego: użytkownik przy pomocy budowy graficznego schematu przetwarzania danych definiuje, w jaki sposób *Clementine* będzie pobierać dane, eksplorować i prezentować wyniki.

8.4 Data Mining Suite, Information Discovery

Data Mining Suite przeznaczony jest do odkrywania wiedzy w bardzo dużych zbiorach danych. Automatycznie znajduje reguły, schematy i anomalie w bazach danych. Proces odkrywania wiedzy może przebiegać automatycznie, bądź też może być nadzorowany i kierowany przez użytkownika. System buduje raporty w języku naturalnym. Dodatkowo, środowisko wyposażone jest w moduł *Predictive Modeler*, służący do predykcji na podstawie odkrytych reguł i schematów. Wspierane są następujące techniki eksploracji danych: klasyfikacja, klastering, odkrywanie charakterystyk, analiza zależności, wykrywanie odchyleń. *Data Mining Suite* korzysta z baz danych poprzez interfejs SQL. *Information Discovery* jest pierwszym partnerem *Oracle*, którego produkty służące do eksploracji danych będą integrowane z serwerem *Oracle Express* [[8]].

9. Bibliografia

[1] „**A Database Perspective on Knowledge Discovery”**, Imielinski T., Manilla H., , Communications of the ACM, Vol. 39, No. 11, Nov. 1996

[2] **„A New SQL-like Operator for Mining Association Rules”**, Meo R., Psaila G., Ceri S., , Proc. of the 22nd VLDB Conference, Bombay, India, 1996,

[3] **„Advances in Knowledge Discovery and Data Mining”**, ed. Fayyad U.M., Piatetsky-Shapiro G., Smyth P., Uthurusamy R., MIT Press, 1996

[4] **„DBMiner: A System for Mining Knowledge in Large Relational Databases”**, Han J., Fu Y., Wang W., Chiang J., Gong W., Koperski K., Li D., Lu Y., Rajan A., Stefanovic N., Xia B., Zaiane O.R., , Proc. Int'l Conf. Data Mining and Knowledge Discovery, Portland, Oregon, August 1996,

[5] **„Discovery board application programming interface and query language for database mining”** Imielinski T., Virmani A., Abdulghani A., Proc. of KDD96, Portland, Oregon, August 1996,

[6] **„IBM Digs Deep for Data Mining ‘Gold’”**, http://www.software.ibm.com/data/intelli-mine/factsheet.html

[7] **„Mine Set - Product Overview”**, http://www.sgi.com/Products/software/MineSet

[8] **„Oracle Adds Data Mining, NT Software And Web Charting Vendors To Warehouse Technology Initiative”**, Press Release, http://www.oracle.com/corporate/press/html/PR100896.153346.html,

[9] **„SQL-like language for database mining”**, Morzy T., Zakrzewicz M., Proc. of the First East-European Symposium on Advances in Databases and Information Systems - ADBIS'97, St. Petersburg, 1997

[10] **„The Clementine Data Mining Tool”**, http://www.isl.co.uk/toolkit.html