STUDIA INFORMATICA 2011
Volume 32 Number 2B (97)

Adam PELIKANT, Anna KOWALCZYK-NIEWIADOMY, Dominik NIEWIADOMY
Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych

ALGORYTM ETYKIETOWANIA ANALIZUJĄCY ROZMYTE ZAPYTANIA W METAJĘZYKU NATURALNYM

Streszczenie. Przedmiotem niniejszego artykułu jest pozyskiwanie nieprecyzyjnych informacji z bazy danych przy wykorzystaniu autorskiego algorytmu etykietowania, wykorzystującego metody sztucznej inteligencji. Za pomocą rozmytych algorytmów grupowania i automatycznego generowania funkcji przynależności analizowane są statystyki ruchu na witrynie WWW. Zastosowanie algorytmu etykietowania pozwoliło na uzyskanie odpowiedzi na zapytanie sformułowane w metajęzyku naturalnym.

Słowa kluczowe: logika rozmyta, grupowanie, zapytania rozmyte.

LABELING METHOD FOR METALANGUAGE QUERIES ANALYSIS

Summary. This paper presents a novel idea of gaining imprecise information from relational database systems. Concernment of investigation rise fact that such kind of processing is not supported by any commercial database system. These researches illustrate a combination of database technology and fuzzy logic. The final aim is to develop a fuzzy querying system based on meta-natural language.

Keywords: fuzzy logic, clustering, grouping, fuzzy queries

1. Wprowadzenie

Obecnie tradycyjne systemy zarządzania bazami danych projektowane są pod kątem efektywnego przechowywania danych oraz szybkiego i wygodnego dostępu do precyzyjnych informacji. Zastosowanie bazodanowej logiki trójwartościowej ogranicza matematyczną interpretację niejednoznaczności i jest dalekie od naturalnego opisu zjawisk. Wraz z intensywnym rozwojem systemów bazodanowych, istnieje potrzeba jak najwierniejszego modelowania

świata rzeczywistego, a co za tym idzie rośnie zapotrzebowanie na rozszerzenie tradycyjnego języka zapytań o nowe możliwości. Poniższe opracowanie stanowi nowatorskie podejście w dziedzinie pozyskiwania nieprecyzyjnych informacji zgromadzonych w systemach zarządzania bazami danych. Celem badań jest opracowanie oprogramowania, które pozwoli na przetwarzanie zapytań z elementami języka naturalnego. Złożony mechanizm przetwarzania danych wykorzystuje znane algorytmy rozmytego grupowania danych oraz autorskie mechanizmy prowadzące do otrzymania funkcji przynależności i zbiorów rozmytych w sposób automatyczny, na podstawie niejednorodnej dystrybucji danych. Algorytm etykietowania, wykorzystujący elementy sztucznej inteligencji, pozwala na powiązanie wyników przetwarzania z wybranymi pojęciami języka naturalnego.

Dotychczasowe rozwiązania, oparte na teorii zbiorów rozmytych, mają silne ograniczenia na etapie konstruowania zbiorów rozmytych. Systemy opisane w [2, 3, 4] zasługują w tym miejscu na uwagę, jednak wymagają one współpracy z ekspertem dziedzinowym, który decyduje o kształcie zbiorów i stopniu rozmycia. Podstawową ideą opisywanych badań jest założenie możliwości automatycznego określenia funkcji przynależności na podstawie rzeczywistego rozkładu danych, które wykazują naturalną tendencję do niejednorodnej dystrybucji. Zastosowanie algorytmów klasyfikacji bez nauczyciela pozwala na wykrycie takiej nierównomierności, a zatem automatyczne określenie liczby zbiorów rozmytych oraz opisujących je funkcji przynależności. W końcowym efekcie pozwoli to na implementację systemu generującego odpowiedź na niejednoznacznie zadane zapytania.

2. Pozyskanie danych wejściowych

Na potrzeby badań utworzony został testowy serwis WWW. Statystyki ruchu witryny okazały się dobrym źródłem danych wejściowych. Jako hurtownię danych statystycznych wykorzystano narzędzie Google Analytics. **Google Analytics** jest darmowym, internetowym narzędziem do analizy statystyk serwisów WWW, udostępnianym przez firmę Google Inc. Jego główne cechy to:

- generowanie raportów związanych ze zbieraniem danych dotyczących ruchu internetowego,
- możliwość tworzenia segmentów użytkowników według źródła ruchu lub zachowania w serwisie,
- eksportowanie raportów w kilku formatach (m.in. CSV oraz XML),
- łatwe integrowanie konta z kontami AdWords,
- integracja z aplikacjami e-commerce,
- dostępność API,

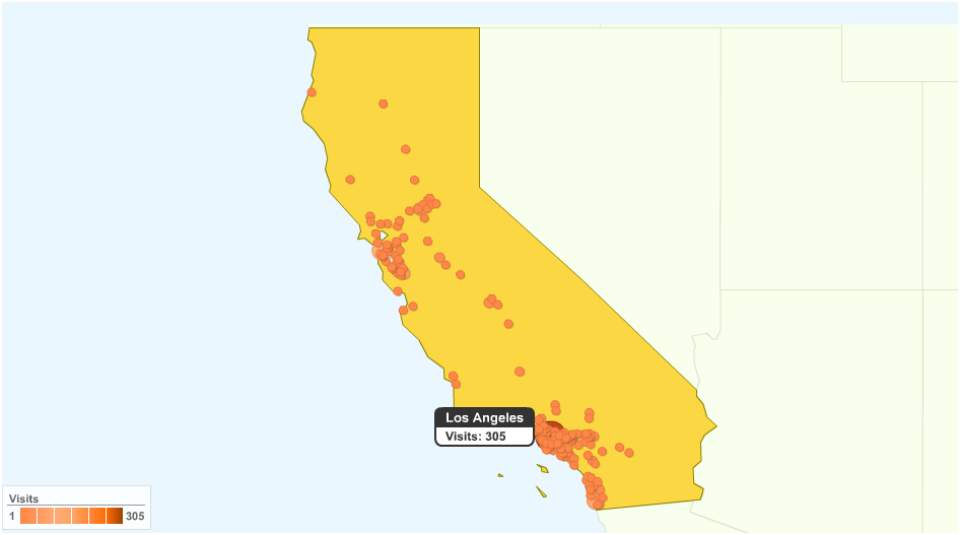
- jednoczesne zbieranie danych z nieograniczonej liczby stron internetowych (jedynym ograniczeniem jest limit 5 milionów odsłon na miesiąc dla osób, które nie korzystają z programu AdWords).

Jednym z elementów badań było przygotowanie oprogramowania pozwalającego na komunikację z narzędziem Google Analytics. Dzięki wykorzystaniu języka programowania Java, Java Google Api (JGA) oraz JDBC, zapewniono integrację z bazą danych Oracle. Prace obejmowały:

- implementację oddzielnego modułu integracji danych z wykorzystaniem Java 6 oraz Google API Java Client,
- zaprojektowanie schematu bazy danych pod kątem wykorzystania do docelowego przetwarzania danych,
- dynamiczne tworzenie tabel opartych na danych pochodzących z zapytań do GA,
- zapytania o n-wymiarów i m-metryk, dynamicznie tworzące tabele znormalizowane w relacyjnej bazie danych.

Przykładowe statystyki ruchu na stronie WWW udostępnione przez GA to:

- liczba wizyt (sumaryczna i unique users),
- liczba odsłon,
- średnia liczba odsłon w trakcie pojedynczej wizyty,
- procent odrzuceń (wejście z 1 odsłoną),
- średni czas spędzony na stronie,
- lojalność użytkowników (procent nowych wejść).



Rys. 1. Graficzny podgląd statystyki Visits dla stanu Kalifornia
Fig. 1. Graphic view of visits statistic for California State

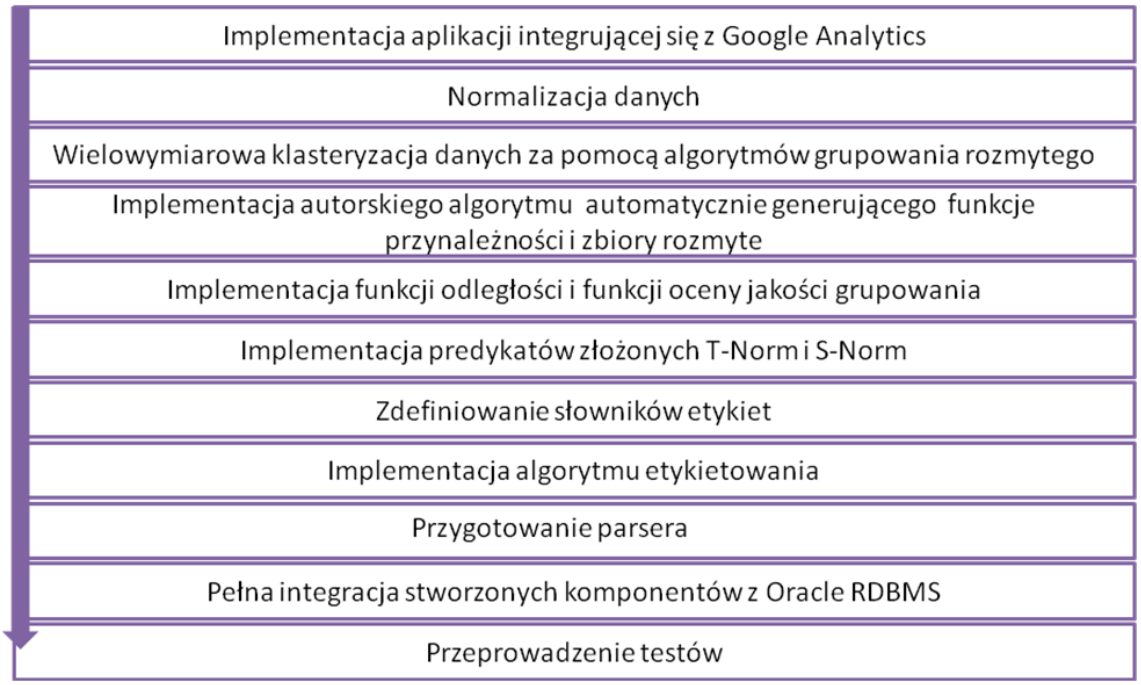
Powyższe statystyki możemy analizować w następujących wymiarach:

- data (godzina, dzień, miesiąc, rok),

- lokalizacja – źródło odwiedzin (kontynent, subkontynent, państwo, region, miasto, szerokość i długość geograficzna),
- typ, wersja i parametry przeglądarki WWW (IE, FF itd.),
- urządzenie, z jakiego nastąpiło wejście na stronę (typ komputera, urządzenia mobilnego, system operacyjny),
- parametry ekranu (liczba kolorów, rozdzielczość),
- język.

3. Prace badawcze

Kolejne etapy badań zaprezentowano na rys. 2. Mając zbiór danych wejściowych w postaci znormalizowanych tabel, w pierwszej kolejności ekspert dziedzinowy podejmuje decyzję o liczbie i rodzaju analizowanych atrybutów.



Rys. 2. Etapy badań
Fig. 2. Research steps

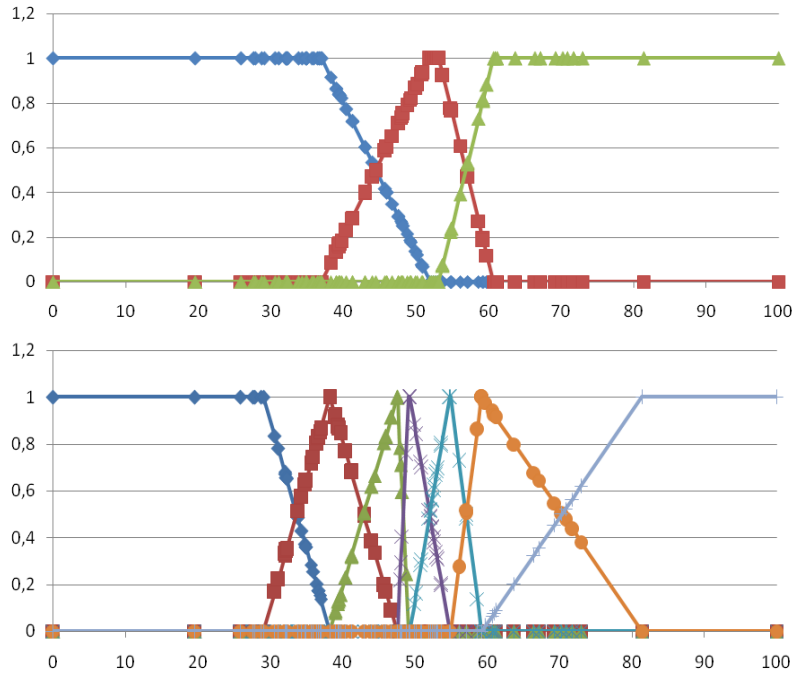
Dane wejściowe podlegają normalizacji do przedziału domkniętego <0, 100> liczb rzeczywistych. Dzięki temu procesowi eliminujemy problem skali wartości ujemnych i zapewniamy integralność na poziomie generowania zbiorów rozmytych. Standaryzacja dotyczy zarówno danych wejściowych, jak i zakresu etykiet. Dzięki temu możliwy jest procentowy przydział etykiet i zapewnienie bezkontekstowości sformułowań, takich jak „wysoka temperatura" rozpatrywana, na przykład w kontekście pogody, temperatury gotowanej wody czy wytopu metali.

3.1. Wielowymiarowe przetwarzanie danych i automatyczne generowanie funkcji przynależności

Z punktu widzenia jakości systemu, najistotniejszym etapem badań jest prawidłowy proces generowania zbiorów rozmytych i funkcji przynależności [5, 6, 7]. Proces ten składa się z dwóch etapów. Pierwszy polega na rozmyciu danych za pomocą algorytmów wielowymiarowego grupowania k-means i mountain clustering. Drugi etap wykorzystuje autorski algorytm generowania funkcji przynależności i zbiorów rozmytych. W wyniku otrzymujemy trapezowy i trójkątny obraz zbiorów rozmytych, który jest analizowany przez moduł oceny jakości grupowania [8, 9]. Przykładowy wynik zaprezentowano rys. 3.

a)

b)



Rys. 3. Zbiory rozmyte uzyskane: a) algorytmem k-means dla k=3 (a) k=7 (b), b) autorskim algorytmem automatycznego generowania funkcji przynależności
Fig. 3. Exemplary fuzzy sets generated by: a) k-means algorithm, b) automatic membership function generator

3.2. Moduł etykietowania

Niezbędnym elementem w procesie przetwarzania zapytania rozmytego jest algorytm analizujący i przetwarzający to zapytanie. Zadaniem algorytmu etykietującego jest prawidłowy przydział etykiet do zbiorów rozmytych, uzyskanych uprzednio w procesie automatycznego generowania funkcji przynależności. W procesie etykietowania można wyróżnić następujące kroki:

1. Zdefiniowanie słowników etykiet w postaci zestawów etykiet z odpowiednim stopniowaniem „siły” każdej etykiety.
2. Skojarzenie zestawów etykiet z konkretnymi atrybutami, pamiętając, że etykiety mogą być wykorzystywane do pracy z więcej niż jednym atrybutem.

3. Ustalenie przedziałów, w jakich dana etykieta ma się zawierać, przy czym etykiety mogą podlegać nierównomiernemu rozkładowi. Ponadto, algorytm dopasowujący etykiety do grup musi reagować na poniższe sytuacje:
   - nadmiar liczby etykiet względem liczby klastrów,
   - nadmiar liczby klastrów względem liczby etykiet,
   - równa liczba etykiet i klastrów.
4. Przygotowanie parsera dokonującego analizy zapytania i automatycznie dobierającego etykiety do atrybutów. Dodatkowo parser ten uruchamia moduły wyliczające przydział danych do poszczególnych klastrów, co pozwala na wygenerowanie wyniku zapytania.

Rozważając rozkład etykiet wyróżniamy kilka przypadków:

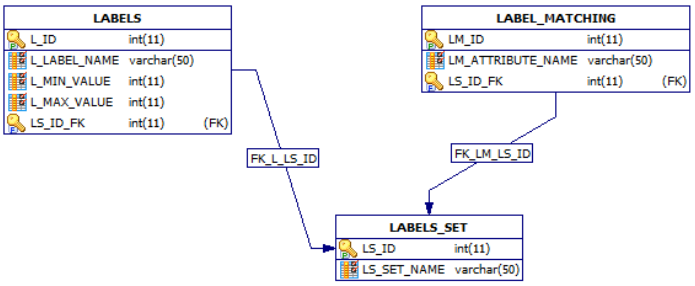
- etykiety rozłożone równomiernie bez zachodzenia na siebie,
- etykiety rozłożone nierównomiernie (wiedza eksperta) bez zachodzenia na siebie,
- etykiety rozłożone równomiernie z zakładką,
- etykiety rozłożone nierównomiernie (wiedza eksperta) z zakładką.

3.2.1. Słowniki etykiet

Definicja słowników etykiet została oparta na strukturze następujących tabel:

- LABELS – tabela zawiera definicje wszystkich etykiet,
- LABELS_SET – tabela zawiera etykiety pogrupowane w zestawy,
- LABEL_MATCHING – tabela pozwala na skojarzenie zestawu etykiet z konkretnym atrybutem.

  Graficzne zależności między tabelami ilustruje rys. 4.



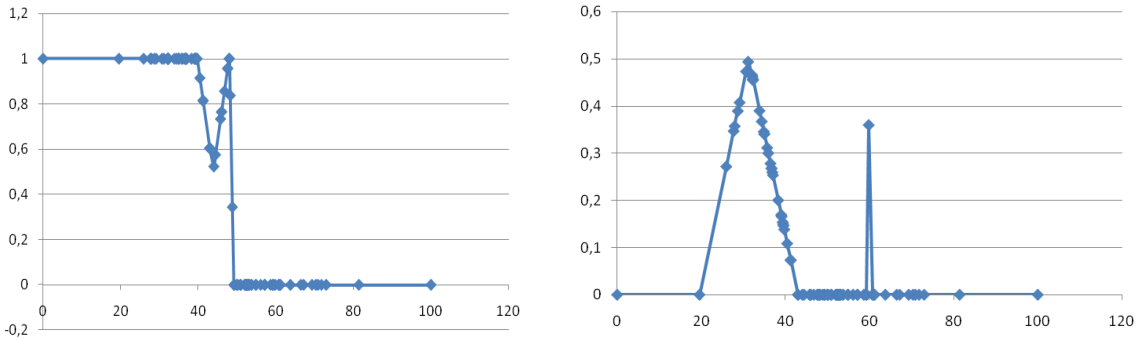
Rys. 4. Struktura tabel dla modułu etykietowania
Fig. 4. Tables structure used by labeling module

Celem projektu jest uzyskanie odpowiedzi na zapytanie o przykładowym kształcie:

```
SELECT * FROM TABELA FUZZYWHERE 'bardzo mały' odwiedzin AND 'mały' odwiedzin OR
'duży' współczynnik_odrzuceń;
```

Przykładowe wyrażenia logiczne w warunku *fuzzywhere* zostały przedstawione poniżej wraz z graficzną prezentacją na rys. 5 dla poniższych warunków:

- *'bardzo mały'* timeonsite OR *'mały'* timeonsite,
- *'bardzo mały'* timeonsite AND *'mały'* timeonsite LUB *'wysoki'* timeonsite AND *'bardzo wysoki'* timeonsite.



Rys. 5. Wynik operacji logicznych na zbiorach rozmytych dla atrybutu timeonsite
Fig. 5. Result of logical operations on timeonsite fuzzy sets

4. Podsumowanie

Na podstawie rzeczywistej dystrybucji danych udało się zdefiniować algorytmy automatycznie generujące zbiory rozmyte. W tym celu wykorzystano rozmyte algorytmy grupowania, a także zastosowano autorskie podejście dążące do generowania funkcji przynależności niezależnie od skali danych wejściowych. Dzięki zastosowaniu normalizacji zapewniono bezkontekstowość zagadnienia oraz jego uniwersalność. Ponadto, zaproponowano mechanizm w pełni zautomatyzowanego etykietowania, wykorzystującego elementy sztucznej inteligencji, składającego się ze słowników etykiet, parsera zapytań oraz algorytmu przydzielania etykiet. Dzięki takiemu podejściu udało się uzyskać odpowiedź na złożone logicznie zapytanie, wykorzystujące elementy języka naturalnego. Pełna implementacja tak opisanego problemu stanowi wartościowy element w dziedzinie baz danych, co dowodzi zasadności prowadzenia dalszych badań.

BIBLIOGRAFIA

1. Zadeh L.A.: Fuzzy sets. Information and Control, 1965.
2. González C., Tineo L., Galindo J.: Fuzzy Database Languages Integration using Expressive Power. Fifth International Conference on Fuzzy Systems and Knowledge Discovery.

3. González C., Goncalves M., Tineo L.: A New Upgrade to SQLf: Towards a Standard in Fuzzy Databases. 20th International Workshop on Database and Expert Systems Application, IEEE Computer Society, DOI 10.1109/DEXA.2009.35.
4. Kacprzyk J., Zadrozny S.: FQUERY for Access: Fuzzy Querying for Windows-Based DBMS, [in:] Bosc P., Kacprzyk J. (eds.): Fuzziness in Database Management Systems. Physica-Verlag, 1995, s. 415÷433.
5. Kowalczyk A., Pelikant A.: Fuzzy Clustering in Relational Databases. XII International Conference-System Modelling and Control, 2007.
6. Kowalczyk A., Pelikant A.: Implementation of automatically generated membership functions based on grouping algorithms. The International Conference on Computer as a tool, 2007.
7. Kowalczyk A., Pelikant A.: Fuzzy queries in relational databases. XIII International Conference-System Modelling and Control 2009 (publikacja pokonferencyjna w JACS 2010).
8. Fraley C., Raftery A.: How many clusters? which clustering method? Answers via model-based cluster analysis. The Computer Journal, Vol. 41, No. 8, 1998, s. 578÷588.
9. Schwartz G.: Estimating the dimension of a model. The Annals of Statistics 6, 1978.
10. Kowalczyk-Niewiadomy A, Pelikant A.: Zagadnienie grupowania w kontekście budowania zapytań rozmytych. Bazy Danych: Rozwój metod i technologii, tom 1. WKiŁ, Gliwice 2008, s. 175÷186.

Recenzenci: Dr inż. Małgorzata Bach
Prof. dr hab. inż. Alicja Wakulicz-Deja

Wpłynęło do Redakcji 30 stycznia 2011 r.

Abstract

This paper, takes into consideration the problem of retrieving ambiguous and imprecise information from relational database system. As the fact, traditional SQL does not provide any essential mechanism for solving such issues. In recent years, fuzzy SQL language that allows making flexible queries has become a very interesting object of research. Unfortunately in most cases the implementation of fuzzy sets theory is based on one constant threshold and depends strictly on experts decision [2, 3, 4].

This article presents the novel way of gaining imprecise and incomplete information from database. The idea based on fuzzy clustering methods provides an effective tool for fuzzy sets

generation and gaining fuzzy query results from database system automatically. The most important points of the idea are the data normalization, fuzzy clustering algorithms and clustering quality measurement methods [8, 9]. What is more, we are able to generate membership functions automatically by means of clustering. Presented conception of labeling mechanism is to enable getting satisfactory result for query written in meta natural language. It is worth mentioning that the results do not depend on context so the solution is universal. Although some methods and technical concepts need to be extended and optimized, a fuzzy clustering and classification querying approach remains effective.

Adresy

Adam PELIKANT: Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych, ul. Stefanowskiego 18/22, 90-924 Łódź, Polska, APelikan@p.lodz.pl.

Anna KOWALCZYK-NIEWIADOMY: Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych, ul. Stefanowskiego 18/22, 90-924 Łódź, Polska, Anna.KowalczykNiewiadomy@gmail.com.

Dominik NIEWIADOMY: Politechnika Łódzka, Instytut Mechatroniki i Systemów Informatycznych, ul. Stefanowskiego 18/22, 90-924 Łódź, Polska, Dominik.Niewiadomy@gmail.com.