STUDIA INFORMATICA 2002
Volume 23 Number 4 (51)

Bożena MAŁYSIAK
Politechnika Śląska, Instytut Informatyki

APROKSYMACYJNE ZAPYTANIA DO BAZ DANYCH

Streszczenie. W artykule przedstawiono różne rodzaje aproksymacyjnych zapytań zadawanych do baz danych, takie jak: wektorowe, rozmyte oraz oparte na prawdopodobieństwie. Każdy rodzaj pytań zilustrowano przykładami.

APPROXIMATE QUERIES IN DATABASES

Summary. This article provides an overview of different kind of approximate queries in databases such as: vector queries, probabilistic queries and fuzzy queries. All described kinds of queries are illustrated by examples.

1. Wprowadzenie

W relacyjnym modelu danych podstawową strukturą służącą do organizacji i przechowywania danych jest relacja (tabela). Dane prezentowane użytkownikowi są wyświetlane również w postaci relacji (tabel) [6]. Relacja ma postać tabeli, w której w nagłówku podane są atrybuty relacji (nazwy kolumn), a następne wiersze nazywane są krotkami i zawierają dane dotyczące tej relacji (wartości jej atrybutów).

Analizując stopień spełnienia kryteriów podanych w pytaniu do bazy danych można wyodrębnić kilka rodzajów zapytań [2]:

- **Dokładne** – w których dokładnie są sprecyzowane kryteria wyszukiwania, a w odpowiedzi uzyskuje się zbiór tylko tych krotek, które spełniają podane w zapytaniu kryteria,
- **zakresowe** – w których jako warunki podane są przedziały (zakresy), a w odpowiedzi uzyskuje się tylko te krotki, których wartości atrybutów mieszczą się w zdefiniowanych przez użytkownika przedziałach,

– **aproksymacyjne** - w odpowiedzi na tak zadane pytania uzyskuje się zbiór krotek spełniających w zadanym stopniu kryteria sprecyzowane w zapytaniu. Zapytania aproksymacyjne wymagają zdefiniowania tzw. funkcji charakterystycznej, która określa, w jakim stopniu wyszukiwana krotka jest zgodna z kryteriami podanymi w zapytaniu.

Ze względu na rodzaj stosowanej funkcji charakterystycznej wyróżnić można następujące rodzaje zapytań aproksymacyjnych:

– zapytania wektorowe (vector queries), w których funkcją charakterystyczną jest funkcja podobieństwa dwóch wektorów,
– zapytania oparte na prawdopodobieństwie (probabilistic queries), w których funkcja charakterystyczna określona jest jako prawdopodobieństwo, z jakim krotka związana jest z pytaniem,
– zapytania rozmyte (fuzzy queries), w których funkcją charakterystyczną jest funkcja określająca stopień przynależności (dopasowania) krotki do kryteriów zapytania,
– zapytania w języku naturalnym (natural language queries).

W tej pracy autorka skupiła się przede wszystkim na przedstawieniu zapytań wektorowych, opartych na podobieństwie dwóch wektorów, zapytań opartych na prawdopodobieństwie, jak również na pytaniach rozmytych, dla których funkcja charakterystyczna przedstawiona jest w postaci funkcji przynależności krotki do zdefiniowanego w pytaniu podzbioru rozmytego. Wszystkie omówione rodzaje zapytań zilustrowano przykładami.

2. Pytania wektorowe

Pytania wektorowe, w których funkcją charakterystyczną jest funkcja podobieństwa dwóch wektorów, są często wykorzystywane w bazach danych zawierających dane posiadające wiele atrybutów. Zbiory krotek są odwzorowywane do takiej reprezentacji, jaka jest podana w zapytaniu. W tym przypadku krotki reprezentowane są w postaci **wektorów** w n-wymiarowej przestrzeni cech.

Pytania tego typu często stosowane są w bazach danych zawierających dokumenty tekstowe (dokumenty). Dokumenty takie przedstawiane są zwykle w postaci uporządkowanej listy wyrazów [2].

Najprostszym modelem opisu dokumentu jest przedstawienie go za pomocą wektora, którego składowe przyjmują tylko liczby binarne {0, 1}: 1 – w przypadku, gdy szukany wyraz występuje w dokumencie, 0 – gdy nie występuje. Ilustruje to przykład 1.

Przykład 1

Jeśli wyrazy, które są szukane, to zbiór: *{komputer, informatyka, dane, pytania}*, a dokument w bazie danych zawiera spośród nich tylko wyrazy *informatyka* i *pytania,* to będzie on reprezentowany jako wektor [0, 1, 0, 1].

Inna nieco rozszerzona metoda opisu może opierać się na **wektorze wag,** którego składnikami są wartości wag związanych z wystąpieniem wyrazu w dokumencie. Ilustruje to przykład 2.

Przykład 2

Jeśli wyrazy, które są szukane, to: *{komputer, informatyka, dane, pytania}* wraz z przypisanymi do nich wagami [20, 15, 40, 25], a dokument zawiera spośród nich tylko wyrazy *informatyka* i *pytania,* to dokument ten będzie reprezentowany jako wektor wag [0, 15, 0, 25], a podobieństwo między dokumentem podanym w pytaniu a wyszukiwanym dokumentem w bazie można wyznaczyć jako sumę wag, np. 15 + 25 = 40.

Wagi mogą być również związane z liczbą wystąpień wyrazu w dokumencie (im częściej wyraz występuje, tym ma większą wagę). Użytkownik może również definiować wagi według swoich kryteriów.

Wyszukiwanie oparte na **funkcji podobieństwa** pozwala na wybranie lub odrzucenie pewnych krotek z przeszukiwanego zbioru krotek na podstawie obliczonej wartości ich podobieństwa do wzorca krotki wynikającego z kryterium zawartego w pytaniu.

Z tego punktu widzenia można wyświetlać wynik zapytania w postaci:

- pewnej liczby (określanej przez użytkownika) krotek najbardziej podobnych do wzorca zadanego w pytaniu,
- wszystkich krotek, których wartości podobieństwa do wzorca przekraczają pewien, ustalony przez użytkownika próg,
- wszystkich krotek posortowanych według ich wartości podobieństwa względem wzorca.

W zależności od możliwości zastosowania określonej metody dostępu krotki mogą być reprezentowane przez [1]:

- **wektory w m-wymiarowej przestrzeni cech** (model VSM – Vector Space Model); podobieństwo tych wektorów jest określane za pomocą funkcji podobieństwa dwóch wektorów przekształcanej następnie w funkcję odległości,
- **funkcję odległości** (MSM – Metric Space Model), umożliwiającą określenie odległości między krotkami w bazie. Taka reprezentacja stosowana jest, gdy trudne jest bądź prawie niemożliwe wyodrębnienie cech charakteryzujących krotki.

Miary podobieństwa mogą być związane z pojęciem **odległości** (obiekty położone blisko siebie w przestrzeni wektorowej są do siebie bardziej podobne) lub z pojęciem **miary kątowej** (obiekty położone w tym samym kierunku są blisko związane).

Miara cosinusowa ***sim(d1, d2)*** – określa podobieństwo między dwiema krotkami, które wyznaczane jest ze wzoru na cosinus kąta między dwoma niezerowymi wektorami:

$$sim(d_1, d_2) = \cos(\vec{u}_1, \vec{u}_2) = \frac{\vec{u}_1 \circ \vec{u}_2}{\|\vec{u}_1\| \cdot \|\vec{u}_2\|}, \quad (1)$$

gdzie:

d_1, d_2 – krotki bazy danych,

$\vec{u}_1, \vec{u}_2$ – wektory reprezentujące krotki d_1, d_2 w przestrzeni wektorowej,

$\circ$ – iloczyn skalarny dwóch wektorów, który może być przedstawiony jako suma iloczynów równoimiennych współrzędnych tych wektorów, wyrażony wzorem:

$$\vec{u}_1 \circ \vec{u}_2 = u_{1x}u_{2x} + u_{1y}u_{2y},$$

$\|*\|$ – norma euklidesowa wektorów (długość wektora), wyrażona wzorem:

$$\|\vec{u}_1\| = \sqrt{u_{1x}^2 + u_{1y}^2}, \quad \|\vec{u}_2\| = \sqrt{u_{2x}^2 + u_{2y}^2}.$$

Za pomocą wzoru (1) przekształca się miarę kątową w miarę z zakresu od 1 (dla najbardziej podobnej krotki) do 0 dla najmniej podobnej.

Odległość Euklidesowa ***O(d1, d2)*** – określa odległość dwóch krotek d_1 i d_2:

$$O(d_1, d_2) = \sqrt{(u_{1x} - u_{2x})^2 + (u_{1y} - u_{2y})^2}. \quad (2)$$

Zastosowanie przedstawionych miar ilustruje przykład 3.

Przykład 3

Dana jest tabela zawierająca dokumenty (tab. 1).

Tabela 1

	Dokumenty
T1	Baśnie, bajki i opowiadania dla dzieci na dzień dobry i dobranoc
T2	Rodzice często kupują dzieciom Baśnie Andersena
T3	Baśnie tysiąca i jednej nocy
T4	Bajki na kasecie video i płycie CD; opowiadania i bajki również dla rodziców
T5	Rodzice dzieciom opowiadają bajki na dobranoc

Przyjęto następujący zbiór wyrazów będących podstawą do wyznaczenia składowych wektorów reprezentujących dokumenty:

{baśnie, bajki, opowiadania, dzieci, dzień dobry, dobranoc, rodzice, kupują, andersena, tysiąca, jednej, nocy, kasecie, video, płycie, CD, opowiadają}

Liczność tego zbioru wynosi: 17

Po alfabetycznym posortowaniu zbiór tych wyrazów jest następujący:

{andersena, bajki, baśnie, CD, dobranoc, dzieci, dzień dobry, jednej, kasecie, kupują, nocy, opowiadania, opowiadają, płycie, rodzice, tysiąca, video}

Każdy dokument z tab. 1 może być przedstawiony jako 17-rozmiarowy wektor, którego składowe określają liczbę wystąpień danego wyrazu w tekście. Przedstawiono to w tab. 2.

Tabela 2

Dokumenty jako 17-rozmiarowe wektory

T1	0	1	1	0	1	1	1	0	0	0	0	1	0	0	0	0	0
T2	1	0	1	0	0	1	0	0	0	1	0	0	0	0	1	0	0
T3	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0
T4	0	2	0	1	0	0	0	0	1	0	0	1	0	1	1	0	1
T5	0	1	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0

Do bazy danych zadawane jest pytanie: *Znaleźć dokumenty, które zawierają wyrazy: bajki, dzieci, rodzice,* reprezentowane przez wektor Q (tab. 3).

Tabela 3

Wektor dla zdefiniowanego pytania

Q	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0

Wartości podobieństwa i odległości zostaną obliczone dla wektora Q reprezentującego pytanie i dwóch wybranych wektorów T4 i T5 reprezentujących dokumenty. Wektory T4 i Q przedstawiono w tab. 4 a T5 i Q w tab. 5.

Tabela 4

Wektory T4 i Q

T4	0	2	0	1	0	0	0	0	1	0	0	1	0	1	1	0	1
Q	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0

Podobieństwo między wektorem dokumentu T4 a wektorem pytania Q, obliczane zgodnie ze wzorem (1), ma wartość:

$$sim(T4,Q) = \frac{2*1+1*1}{\sqrt{4+1+1+1+1+1+1}\sqrt{1+1+1}} = \frac{3}{\sqrt{30}} = 0{,}548,$$

a odległość między tymi wektorami obliczana zgodnie ze wzorem (2) wynosi:

$$O(T4,Q) = \sqrt{(2-1)^2+(1-0)^2+(0-1)^2+(1-0)^2+(1-0)^2+(1-0)^2+(1-1)^2+(1-0)^2} = \sqrt{7} = 2{,}645.$$

Tabela 5

Wektory T5 i Q

T5	0	1	0	0	1	1	0	0	0	0	0	0	1	0	1	0	0
Q	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0

Obliczane w ten sam sposób podobieństwo i odległość między wektorem dokumentu T5 a wektorem pytania Q (tabela 5) wynoszą odpowiednio:

$$sim(T5,Q) = \frac{1*1+1*1+1*1}{\sqrt{1+1+1+1+1}\sqrt{1+1+1}} = \frac{3}{\sqrt{15}} = 0{,}775,$$

$$O(T5,Q) = \sqrt{(1-1)^2+(1-0)^2+(1-1)^2+(1-0)^2+(1-1)^2} = \sqrt{2} = 1{,}414.$$

Z przytoczonych obliczeń wynika, że dokument T5 jest bardziej podobny do zadanego pytania Q niż dokument T4 i odległość między dokumentem T5 a Q jest mniejsza niż między T4 a Q.

3. Pytania oparte na prawdopodobieństwie

W pytaniach opartych na prawdopodobieństwie funkcją wyznaczającą kryterium wyszukiwania jest prawdopodobieństwo określające, w jakim stopniu krotka spełnia kryteria postawione w pytaniu [2]. W odpowiedzi na takie pytania uzyskuje się zbiór krotek, które spełniają kryteria pytania z prawdopodobieństwem większym od progu określonego przez użytkownika.

Założenia:

Baza danych zawiera N krotek, a n z nich spełnia kryteria pytania. Oznacza to, że:

- *P(sp)* - prawdopodobieństwo, że losowo wybrana krotka spełnia kryteria pytania, wynosi: $P(sp) = \frac{n}{N}$,
- *P(nsp)* - prawdopodobieństwo, że losowo wybrana krotka nie spełnia kryteriów pytania, wynosi: $P(nsp) = \frac{N-n}{N}$,
- *P(sp\wybr)* - prawdopodobieństwo, z jakim wybrana krotka spełnia kryteria pytania,
- *P(nsp\wybr)* - prawdopodobieństwo, z jakim wybrana krotka nie spełnia kryteria pytania.

Dalej przyjęto, że zbiór wynikowy powinien zawierać krotki spełniające kryteria pytania z prawdopodobieństwem P(sp\wybr) > 0.5. Wartość progowa 0.5 tego prawdopodobieństwa jest określana potocznie jako kryterium wyświetlania.

Wynika z tego, że krotka pojawi się w zbiorze wynikowym tylko wtedy[2], gdy funkcja $d(wybr) = \frac{P(sp \setminus wybr)}{P(nsp \setminus wybr)}$ przyjmie wartość większą od 1.

Korzystając z definicji prawdopodobieństwa warunkowego:

$P(A \setminus B) = \frac{P(B \setminus A)P(A)}{P(B)}$ otrzymuje się zależność:

$$d(wybr) = \frac{P(sp \setminus wybr)}{P(nsp \setminus wybr)} = \frac{P(wybr \setminus sp)P(sp)P(wybr)}{P(wybr)P(wybr \setminus nsp)P(nsp)} = \frac{P(wybr \setminus sp)P(sp)}{P(wybr \setminus nsp)P(nsp)}. \quad (3)$$

W przypadku bazy danych zawierającej dokumenty, dokument (krotka) reprezentowany jest jako zbiór wyrazów w_1, w_2, ..., w_n. Wyrazy występujące w dokumencie są niezależne, zatem prawdopodobieństwo wybrania dokumentu spełniającego kryteria pytania wynosi:

$$P(wybr \setminus sp) = P(w_1 \setminus sp) * P(w_2 \setminus sp) * ... * P(w_n \setminus sp), \quad (4)$$

natomiast prawdopodobieństwo wybrania dokumentu nie spełniającego kryteriów pytania wynosi:

$$P(wybr \setminus nsp) = P(w_1 \setminus nsp) * P(w_2 \setminus nsp) * ... * P(w_n \setminus nsp). \quad (5)$$

Po obliczeniu prawdopodobieństwa, z jakim dokument jest zgodny z pytaniem, wyznaczana jest wartość funkcji *d(wybr)*, która określi, czy dany dokument ma znaleźć się na liście dokumentów spełniających kryteria pytania, czy też nie. Ilustruje to przykład 4.

Przykład 4

Dla pewnego pytania i określonej bazy danych prawdopodobieństwo, że losowo wybrany dokument spełnia kryteria zapytania, wynosi 0,2.

Wybrany dokument D składa się z 5 wyrazów, dla których zostały podane przykładowe dane określające prawdopodobieństwa wystąpienia wyrazów w dokumentach spełniających *P(w, sp)* oraz w dokumentach nie spełniających *P(w, nsp)* kryteriów zapytania. Dany wyraz w_i może wystąpić jednocześnie w dokumentach spełniających, jak również w dokumentach nie spełniających kryteriów zapytania. Zatem prawdopodobieństwa $P(w_i, sp)$ i $P(w_i, nsp)$ nie muszą sumować się do 1. Dane te zebrano w tab. 6.

Tabela 6

Wyrazy oraz ich prawdopodobieństwa P(w, sp) i P(w, nsp)

Wyraz	$P(w_i, sp)$	$P(w_i, nsp)$
w_1	0.7	0.4
w_2	0.6	0.2
w_3	0.1	0.8
w_4	0.9	0.5
w_5	0.4	0.3

Prawdopodobieństwo, że losowo wybrany dokument spełnia kryteria zapytania, wynosi $P(sp) = 0{,}2$, zatem prawdopodobieństwo, że losowo wybrany dokument nie spełnia kryteriów zapytania, wynosi $P(nsp) = 1 - 0{,}2 = 0{,}8$

Wyrazy występujące w dokumencie są niezależne, zatem prawdopodobieństwo wybrania dokumentu spełniającego kryteria pytania wynosi:

$$P(wybr \setminus sp) = P(w_1, sp) * P(w_2, sp) * ... * P(w_5, sp) =$$
$$0.7 * 0.6 * 0.1 * 0.9 * 0.4 = 0.01512,$$

natomiast prawdopodobieństwo wybrania dokumentu nie spełniającego kryteriów pytania wynosi:

$$P(wybr \setminus nsp) = P(w_1, nsp) * P(w_2, nsp) * ... * P(w_5, nsp) =$$
$$0.4 * 0.2 * 0.8 * 0.5 * 0.3 = 0.0096,$$

a więc wartość funkcji obliczona ze wzoru (3) wynosi:

$$d(wybr) = d(D) = \frac{P(wybr \setminus sp) * P(sp)}{P(wybr \setminus nsp) * P(nsp)} = \frac{0.01512 * 0.2}{0.0096 * 0.8} = \frac{0.003024}{0.00768} = 0.39375.$$

Zatem dokument D rozpatrywany w przykładzie nie powinien znaleźć się w zbiorze krotek spełniających kryteria wyszukiwania, ponieważ wartość funkcji d jest mniejsza niż 1.

4. Pytania rozmyte

W klasycznych pytaniach zadawanych do bazy danych krotka spełniała kryteria podane w zapytaniu lub nie, a funkcja charakterystyczna przyjmowała dwie wartości: 1 – gdy krotka spełniała warunki pytania i należała do zbioru wynikowego odpowiedzi na pytanie i 0 w przeciwnym przypadku.

W pytaniach rozmytych wykorzystano elementy teorii zbiorów rozmytych, a funkcją charakterystyczną określającą, w jakim stopniu krotka spełnia kryteria wyszukiwania podane w pytaniu, jest funkcja przynależności, która może przyjmować wartości z przedziału <0, 1>.

Definicja zbioru rozmytego, zaproponowana przez L.A. Zadeha [4] brzmi:

Zbiór rozmyty (Fuzzy set) **A** to:

zbiór par:

$A = \{(\mu_A^*(x), x)\}$, dla każdego $x \in X$,

gdzie:

μ_A – funkcja przynależności zbioru rozmytego A, która każdemu elementowi zbioru $x \in X$ przypisuje stopień jego przynależności $\mu_A^*(x)$ do zbioru A, przy czym: $\mu_A(x) \in [0,1]$.

Zarówno w bazach danych, jak i w pytaniach zadawanych do nich atrybuty mogą przyjmować wartości rozmyte jak i dokładne. Z tego punktu widzenia można wyróżnić kilka przypadków porównań wartości atrybutów z kryterium pytania [5]:

- obie wartości są dokładne; wtedy stopień przynależności przyjmuje wartości 0 (gdy wartości są różne) lub 1 (gdy wartości są jednakowe),
- jedna z wartości jest dokładna druga rozmyta; w tym przypadku element rozmyty reprezentowany jest jako podzbiór rozmyty, dla którego zdefiniowana jest funkcja przynależności, a wartość dokładna reprezentowana jest przez linię pionową; punkt przecięcia funkcji przynależności i linii pionowej wyznacza stopień spełnienia (dopasowania) kryteriów τ,
- obie wartości są rozmyte; wtedy stopień spełnienia (dopasowania) - τ między dwoma elementami rozmytymi w_1 i w_2 (reprezentowanymi przez dwa podzbiory rozmyte, dla których określone są funkcje przynależności) wyznaczany jest jako przecięcie dwóch funkcji przynależności. W przypadku, gdy jest wiele punktów przecięcia, brany jest pod uwagę ten o największej wartości.

W literaturze stosuje się określenia: stopień przynależności, stopień dopasowania i stopień spełnienia jako równoważne. W pracy również terminy te są tak traktowane.

Wymienione przypadki dotyczą porównań jednego atrybutu. Klauzula *WHERE* w instrukcji *SQL* może zawierać wielokrotne porównania połączone poprzez operatory *AND, OR* lub *NOT.*

Gdy łącznikiem w klauzuli *WHERE* jest *AND,* stopień dopasowania wyznaczany jest jako wartość minimalna spośród wszystkich wyznaczonych wcześniej wartości. Jeżeli łącznikiem jest OR, stopień dopasowania jest wyznaczany jako wartość maksymalna spośród wszystkich obliczonych stopni przynależności. Natomiast, gdy łącznikiem jest *NOT*, wartość stopnia dopasowania wyznaczana jest w wyniku odjęcia od 1 wartości stopnia dopasowania.

W pewnych sytuacjach krotka może należeć do relacji z podanym już wcześniej stopniem dopasowania μ, a ponadto można wyznaczyć wartość stopnia dopasowania μ_1 tej krotki do zadanego pytania rozmytego. W tym wypadku całkowity stopień dopasowania tej krotki określony jest jako wartość $min\ \{\mu, \mu_1\}$.

W wyniku wyznaczenia stopni dopasowania względem klauzuli *WHERE* powstaje zbiór krotek wynikowych. Na krotki te w klauzuli *SELECT* nakładane są warunki wycinające część atrybutów wynikowych (te które są wyspecyfikowane w klauzuli *SELECT).*

Proces uzyskiwania dokładnej odpowiedzi na niedokładne (rozmyte) pytanie nazywany jest procesem defuzzyfikacji. Najprostszym sposobem uzyskania odpowiedzi jest znalezienie wartości o maksymalnym stopniu dopasowania, ale stosuje się również inne [3]:

- metodę środka ciężkości,
- metodę środka sum,
- metodę środka sumy ważonej,
- metodę pierwszego z maksimów,
- metodę środkowego maksimum.

Wyznaczanie stopni dopasowania w instrukcji *SELECT*, przedstawia tab. 7 [5]:

Tabela 7

Wyznaczanie stopni dopasowania w instrukcji *SELECT*

WHERE	AND	*Min*
	OR	*Max*
	NOT	$1-\mu$
Krotka ma początkowy stopień dopasowania μ μ_1 wynikający z obliczenia w klauzuli WHERE		*Min* $\{\mu, \mu_1\}$
Klauzula *SELECT*		*Max*

Przykład 5 ilustruje proces wyszukiwania odpowiedzi na niedokładne pytanie kierowane do bazy danych zawierającej dokładne dane.

Przykład 5

Do bazy danych zadawane jest następujące pytanie:

Wypisać nazwiska dobrych i chodzących na zajęcia studentów.

Relację *Studenci* przedstawia tab. 8.

Tabela 8

Studenci

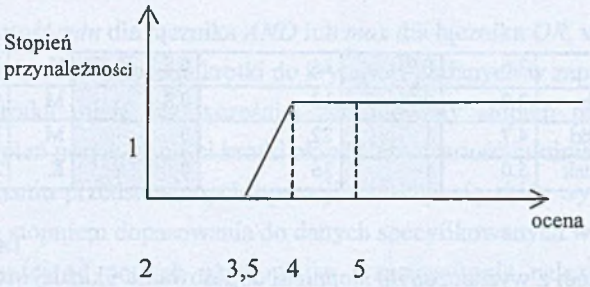
Nr	Imię	Nazwisko	Średnia	Liczba dni nieobecnych	Płeć	Adres
1	Krzysztof	Kowalski	4,3	8	M	Zabrze
2	Katarzyna	Ptak	3.6	3	K	Katowice
3	Marcin	Białas	3.8	8	M	Gliwice
4	Kornel	Jasiński	4.7	22	M	Dąbrowa
5	Hanna	Kogutek	5.0	8	K	Jeziorany

Zdefiniowane są dwa podzbiory rozmyte: *dobry student* oraz *chodzący na zajęcia*.

Oba podzbiory opisane są trapezowymi funkcjami przynależności, przedstawionymi odpowiednio na rys. 1 i rys. 2.

Funkcja przynależności do podzbioru *dobry student* jest opisana wzorem:

$$\mu(x) = \begin{cases} 0 \text{ dla } x \le 3{,}5 \\ \dfrac{x-3{,}5}{0{,}5} \text{ dla } 3{,}5 < x \le 4, \\ 1 \text{ dla } x > 4 \end{cases}$$

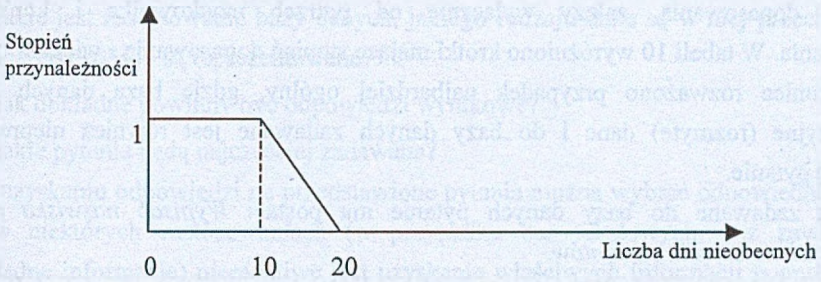


Rys. 1. Funkcja przynależności do podzbioru *dobry student*
Fig. 1. Membership function for fuzzy set *good student*

Funkcja przynależności do podzbioru *chodzący na zajęcia* jest opisana wzorem:

$$\mu(x) = \begin{cases} 1 \text{ dla } x \le 10 \\ \dfrac{20-x}{10} \text{ dla } 10 < x \le 20, \\ 0 \text{ dla } x > 20 \end{cases}$$

Dziedziną zbioru *chodzący na zajęcia* jest liczba dni szkolnych w roku – przedział <0, 200>, rozpatruje się więc liczbę dni nieobecnych.



Rys. 2. Funkcja przynależności do podzbioru *chodzący na zajęcia*
Fig. 2. Membership function for fuzzy set *going on lectures*

Gdy są zdefiniowane funkcje przynależności, należy dla każdej krotki wyznaczyć:

- stopnie przynależności dla każdego atrybutu we frazie *WHERE*,
- spośród obliczonych stopni wyznaczyć wartość minimalną, określającą stopień - τ dopasowania krotki do kryteriów zapytania,
- wyznaczyć zbiór wartości wynikowych w zależności od wyboru metody defuzzyfikacyjnej.

Wyniki kolejnych etapów tego procesu przedstawione są w tab. 9 i 10.

Tabela 9

Relacja Studenci z obliczonymi stopniami przynależności

Nr	Imię	Nazwisko	Średnia	μ(dobry student)	Liczba dni nieobecnych	μ(chodzacy na zajęcia)	Płeć	Adres
1	Krzysztof	Kowalski	4,3	1	8	1	M	Zabrze

cd. tab. 9

2	Katarzyna	Ptak	3.6	0.2	15	0.5	K	Katowice
3	Marcin	Białas	3.8	0.6	12	0.8	M	Gliwice
4	Kornel	Jasiński	4.7	1	22	0	M	Dąbrowa
5	Hanna	Kogutek	5.0	1	36	0	K	Jeziorany

Tabela 10

Relacja Studenci z wyznaczonym stopniem dopasowania τ każdej krotki

Nr	Imię	Nazwisko	Średnia	τ	Liczba dni nieobecnych	Płeć	Adres
1	Krzysztof	Kowalski	4,3	1	8	M	Zabrze
2	Katarzyna	Ptak	3.6	0.2	15	K	Katowice
3	Marcin	Białas	3.8	0.6	12	M	Gliwice
4	Kornel	Jasiński	4.7	0	22	M	Dąbrowa
5	Hanna	Kogutek	5.0	0	36	K	Jeziorany

To, które spośród krotek spełniających kryteria zapytania zostaną wyświetlone, czy wszystkie ze stopniami dopasowania większymi od 0, czy tylko krotka z maksymalnym stopniem dopasowania, zależy wyłącznie od potrzeb użytkownika i konkretnego zastosowania. W tabeli 10 wyróżniono krotki mające stopień dopasowania τ większy od 0.5.

Na koniec rozważono przypadek najbardziej ogólny, gdzie baza danych zawiera nieprecyzyjne (rozmyte) dane i do bazy danych zadawane jest również nieprecyzyjne (rozmyte) pytanie.

Niech zadawane do bazy danych pytanie ma postać: *Wypisać nazwiska dobrych i chodzących na zajęcia studentów.*

W bazie danych w relacji Studenci znajdują się atrybuty rozmyte: *średnia* i *liczba dni nieobecności*. Wartości tych atrybutów mogą być wyrażone poprzez wyrazy rozmyte odpowiednio: około 4, całkiem dobra, około 15, kilkanaście, dwadzieścia kilka itd.

Każdy z wprowadzonych wyrazów rozmytych reprezentowany jest przez podzbiór rozmyty, dla którego jest zdefiniowana funkcja przynależności.

W przypadku tym obie wartości, zarówno ta określona w kryteriach zapytania jak i wartość atrybutu w relacji, są wyrazami rozmytymi.

Stopień dopasowania między kryterium pytania a wartością atrybutu (reprezentowanymi przez dwa podzbiory rozmyte, dla których określone są funkcje przynależności) wyznaczany jest jako przecięcie dwóch funkcji przynależności. Gdy w wyniku wykonania tej operacji jest wiele punktów przecięcia, brany jest pod uwagę ten o największej wartości.

Gdy w zapytaniu występuje więcej kryteriów, proces ten jest powtarzany dla każdego z nich. Następnie rozważa się łącznik, jaki występuje w klauzuli *WHERE* i odpowiednio

wyznacza wartość *min* dla łącznika *AND* lub *max* dla łącznika *OR*, wyznaczając w ten sposób całkowity stopień dopasowania krotki do kryteriów podanych w zapytaniu.

Gdyby krotka miała już wcześniej zdefiniowany stopień przynależności do relacji, całkowity stopień przynależności krotki określałaby wartość minimalna spośród nich.

Po wykonaniu przedstawionych operacji uzyskuje się zbiór wynikowy złożony z krotek z określonym stopniem dopasowania do danych specyfikowanych w zapytaniu.

W zależności od potrzeb użytkownika i zastosowania należy wyświetlić najbardziej reprezentatywne krotki relacji. Jako kryterium można wybrać operator MAX (wybierający wiersze o maksymalnym stopniu dopasowania) lub zastosować inne metody defuzzyfikacji.

5. Podsumowanie

W pracy przedstawiono różne rodzaje aproksymacyjnych zapytań do baz danych. Wybór odpowiedniej metody aproksymacyjnego wyszukiwania nie jest rzeczą prostą. Przed wyborem najwłaściwszego rozwiązania należy odpowiedzieć sobie na kilka pytań:

- jakie jest zastosowanie bazy danych, jakiego rodzaju dane są w niej przechowywane i w jaki sposób są reprezentowane?
- jak dokładne powinny być odpowiedzi wynikowe?
- jakie pytania będą najczęściej zadawane?

Po uzyskaniu odpowiedzi na przedstawione pytania można wybrać odpowiednią metodę, gdyż w niektórych zastosowaniach (w przypadku baz tekstowych, baz zawierających niedokładne informacje) niemożliwe jest uzyskanie właściwych informacji poprzez zadanie dokładnego pytania. Zastosowanie w takich przypadkach odpowiedniej metody aproksymacyjnego wyszukiwania znacznie ułatwia użytkownikowi pracę, a często staje się konieczne.

LITERATURA

1. White D. A., Jain R.: Algorithms and Strategies for Similarity Retrieval. Adres internetowy: http://vision.ucsd.edu.papers/simret/journ1.html.
2. Document and Queries. Adres internetowy: http://www.coe.uncc.edu/~eelkwae/CSC12170Spring2000.
3. Yager R, Filev D.: Podstawy modelowania i sterowania rozmytego. Wydawnictwa Naukowo-Techniczne, Wiley. Warszawa 1995.
4. Zadeh. L. A.: Fuzzy sets, Information and Control 8, 338-353, 1965.

5. Yu C.T., Meng W.: Principles of Database Query Processing for Advanced Applications. Morgan Kaufmann Publishers, Inc., 1998.
6. Ullman J. D., Widom J.: Podstawowy wykład z systemów baz danych. Wydawnictwa Naukowo-Techniczne, Warszawa 2000.

Recenzent: Dr hab. inż. Stanisław Wołek, prof. Pol. Rzeszowskiej

Wpłynęło do Redakcji 27 grudnia 2001 r.

Abstract

In this work approximate methods of object searching in databases are described. This article provides an overview of different kind of queries in databases such as: vector queries, probabilistic queries and fuzzy queries.

The second chapter presents vector queries, where each document is represented by a vector.

In the third chapter probabilistic queries are described. The basic idea of probabilistic matching is to calculate the probability that a document is relevant to a query.

In the next chapter fuzzy queries are presented. Fuzzy queries are based on fuzzy logic, in which a term is considered as a fuzzy set, which has defined own membership function.

All of described kinds of queries are illustrated by examples.