

Prace Naukowe Centrum Obliczeniowego
Politechniki Wrocławskiej

10

Seria: Monografie

2

Zygmunt Mazur

**Modele i modyfikacje rozproszonych systemów
wyszukiwania informacji
opartych na tezaursach z wagami**

Wrocław 1989



PRACE NAUKOWE POLITECHNIKI WROCŁAWSKIEJ

No. 10	Scientific Papers of the Computer Centre of the Technical University of Wrocław	No. 10
Monographs	No. 2	1989

Zygmunt MAZUR

Models and modifications of distributed information
retrieval systems based on thesauri with weights

Prace Naukowe Centrum Obliczeniowego
Politechniki Wrocławskiej

10

Seria:
Monografie

2

Zygmunt Mazur

**Modele i modyfikacje rozproszonych
systemów wyszukiwania informacji
opartych na tezaurusach z wagami**



Wydawnictwo Politechniki Wrocławskiej · Wrocław 1989

Recenzenci

Jacek BAŃKOWSKI

Władysław M. TURSKI

Redaktor naukowy

Witold KOMOROWSKI

Opracowanie redakcyjne i korekta

Janina SURMACZYŃSKA

Pracę częściowo wykonano w ramach CPBR 8.13 „Budowa krajowej akademickiej sieci komputerowej”, cel badawczy nr 52 „Formalne metody specyfikacji i konstrukcji oprogramowania sieciowego”.

© Copyright by Wydawnictwo Politechniki Wrocławskiej, Wrocław 1989

WYDAWNICTWO POLITECHNIKI WROCŁAWSKIEJ

Wybrzeże Wyspiańskiego 27, 50-370 Wrocław

ISSN 0860-7311

Nakład 150 + 75 egz. Ark. wyd. 7,5. Ark. druk. 7³/₈. Papier offset. kl. IV, 70 g.

Oddano do druku w lipcu 1989 r. Druk ukończono w sierpniu 1989 r.

Zakład Graficzny Politechniki Wrocławskiej. Zam. nr 1153/89. Cena zł 450,-

*tezaurus, system wyszukiwania
informacji, rozproszony system,
lista inwersyjna, przestrzeń
dokumentów, sprzężenie zwrotne,
adaptacja systemu.*

ZYGMUNT MAZUR*

MODELE I MODYFIKACJE ROZPROSZONYCH SYSTEMÓW WYSZUKIWANIA INFORMACJI OPARTYCH NA TEZAUROSACH Z WAGAMI

Większość dotychczasowych systemów wyszukiwania informacji i zbiorów danych była tworzona niezależnie od siebie. Chcąc połączyć wiele różnych niezależnych systemów wyszukiwania informacji w jeden duży, rozproszony system użytkowany w sieci komputerowej, należy rozwiązać wiele nowych problemów nie występujących w tradycyjnych centralnych systemach wyszukiwania informacji. W monografii analizuje się mechanizm szukania odpowiedzi w rozproszonych systemach. Rozważa się problem połączenia n lokalnych systemów wyszukiwania informacji. Na bazie lokalnych podsystemów opartych na tezaurusach z wagami zbudowano rozproszony system wyszukiwania informacji i określono strukturę logiczną mechanizmu uzyskiwania globalnej odpowiedzi w rozproszonym systemie na podstawie odpowiedzi z lokalnych podsystemów. Podano modyfikację klasycznej metody list inwersyjnych uwzględniającej relację uogólnienia i wagi deskryptorów w opisach dokumentów. Dokonano przeglądu różnych rozproszonych systemów. Aby móc zaspokajać wymagania użytkowników, zastosowano dynamiczną przestrzeń dokumentów, w której dokumenty mogą zmieniać swoje pozycje zgodnie z wymaganiami użytkowników. Podano geometryczną interpretację rozproszonego systemu. Aktualizacja wag deskryptorów opisów dokumentów na podstawie subiektywnej opinii użytkowników o trafności odpowiedzi pozwala utworzyć opisy lepiej odzwierciedlające rzeczywistość ich treść. Omówiono sposoby modyfikacji opisów dokumentów na podstawie ocen użytkowników o relewantności dokumentów należących do odpowiedzi. Pokazano, że ze zmodyfikowanej przestrzeni dokumentów korzyści czerpią nie tylko ci użytkownicy, którzy brali udział w modyfikowaniu dokumentów. Nowi użytkownicy korzystają z "doświadczeń" zebranych podczas poprzednich poszukiwań. Podano również statystyczne kryterium zmniejszenia redundancji w strukturach wyszukiwawczych oraz oszacowano pesymistyczne koszty kwerend w rozproszonym systemie.

* Centrum Obliczeniowe Politechniki Wrocławskiej, 50-370 Wrocław,
ul. Wybrzeże Wyspiańskiego 27.

"Menon: I w jaki sposób będziesz razem szukał, Sokratesie, tego, o czym w ogóle nie wiesz, co to jest? Bo którą z tych rzeczy, o których nic nie wiesz, weźmiesz przed siebie i zaczniesz szukać? Przecież, gdybyś w sam raz na nią natrafił, to jak będziesz wiedział, że to właśnie to, czegoś nie znalazł?"

Sokrates: Ja rozumiem, co ty chcesz powiedzieć, Menonie. Widzisz, jaką ty sporną myśl wprowadzasz, że nie może człowiek szukać ani tego co zna, ani tego, czego nie zna. Bo jak zna, to przecież nie będzie szukał. Już to zna. Takiemu nie trzeba dopiero szukać. Ani tego, czego nie zna. Bo wtedy nie wie tego nawet, czego ma szukać".

Platon *Menon*

Przekład W. Witwicki,
Warszawa, PWN, 1959, s. 34.

WSTĘP

Przedmiot pracy

Szybki rozwój badań naukowych pociąga za sobą szybki wzrost liczby publikacji naukowych. Obecnie publikuje się na świecie corocznie kilka milionów artykułów z dziedziny nauk przyrodniczych i technik, kilkadziesiąt tysięcy książek, pół miliona opisów patentowych oraz setki tysięcy innych dokumentów naukowych [26]. Fakt podwojenia objętości publikacji naukowo-technicznych co 10-15 lat pozwala wyciągnąć wniosek, że na początku XXI w., gdy liczba nowych informacji wzrośnie dwukrotnie, ogólna liczba publikacji wzrośnie trzydziestokrotnie [40]. Odszukanie w tej ogromnej masie literatury naukowo-technicznej publikacji interesujących czytelników staje się coraz trudniejsze. W związku z tym coraz większego znaczenia nabierają prace nad rozwojem zautomatyzowanych systemów wyszukiwania informacji mające na celu ich doskonalenie. Na doskonalenie i rozwój systemów wpływa nie tylko wzrost zasobu dokumentów, lecz także wzrost liczby, złożoności i pilności kwerend powodowany

nieprzerwanym wzrostem liczby uczonych i specjalistów, a także ilości i złożoności rozwiązywanych przez nich zadań naukowo-technicznych. Dlatego też terminowe, dokładne i kompletne wydanie dokumentów w odpowiedziach na liczne i różnorodne kwerendy staje się zadaniem masowej obsługi, które może być rozwiązane tylko za pomocą odpowiednich środków automatyzacji.

Wyszukiwanie informacji jest procesem odszukania w zbiorze tych dokumentów, które poświęcone są wskazanemu w kwerendzie tematowi lub zawierają niezbędne dla użytkownika fakty i informacje. Można wyróżnić trzy podstawowe rodzaje systemów:

- (a) systemy baz danych,
- (b) systemy wyszukiwania informacji,
- (c) systemy eksperckie.

W tych trzech typach systemów wiele problemów jest wspólnych. Związane są one m.in. z gromadzeniem i przechowywaniem danych oraz wyszukiwaniem pamiętanych zbiorów danych w celu przygotowania odpowiedzi na zapytania użytkowników. W praktyce jednak użytkownicy tych systemów należą do różnych grup mających wiele różnorodnych problemów [2], [39].

1. *Systemy baz danych* używane są do przetwarzania strukturalnych zbiorów danych przedstawianych w postaci dwuwymiarowych tablic i związków oznaczających powiązania pomiędzy różnymi tablicami. Każdy wiersz takiej tablicy może być wykorzystany do przedstawienia rekordu zawartego w zbiorze danych, każda kolumna tablicy identyfikuje pewien atrybut, którego wartości są używane do odróżniania rekordów jeden od drugiego. W szczególności, np. w systemie ewidencji ludności, wiersze tablicy opisują poszczególne osoby, a kolumny zawierają informacje na temat adresów, stanowisk, zarobków pracowników itp. Wyszukiwanie w systemach baz danych polega na znalezieniu tych rekordów, które przyjmują określone wartości w polach odpowiadających atrybutom występujących w zapytaniach użytkowników. Problemy szczególnie interesujące użytkowników systemów zarządzania bazami danych związane są z efektywnym przetwarzaniem struktur reprezentowanych przez tablice [27], [97].

2. *Systemy wyszukiwania informacji* zwykle obejmują obiekty zawierające niesformatowane dane (informacje), aczkolwiek w dokumentach mogą wystąpić również sformatowane atrybuty. Systemy te przeznaczone są do wyszukiwania informacji w zbiorach tych obiektów, które poświęcone są wskazanemu w kwerendzie tematowi lub zawierają niezbędne dla użytkownika fakty i informacje. Wyszukiwanie informacji oparte jest na wykorzystaniu charakterystyk wyszukiwawczych zamiast na czytaniu lub choćby pobieżnym przeglądaniu pełnych tekstów dokumentów. Charakterystyką

wyszukiwawczą dokumentu jest sformułowany według określonych reguł tekst, w którym jest zawarty zasadniczy temat lub przedmiot tego dokumentu. Im krócej sformułowane są charakterystyki wyszukiwania, tym większa jest szybkość wyszukiwania, lecz jednocześnie mniejsza dokładność i kompletność [30], [83]. Jednym z podstawowych problemów dowolnego systemu przechowywania i wyszukiwania informacji jest określenie odpowiedniej struktury informacyjnej umożliwiającej szybki dostęp do dokumentów odpowiadających danej kwerendzie.

3. *Systemy eksperckie zawierają nie tylko dane o obiektach, lecz także zbiory reguł wnioskowania, z których pomocą można wyprowadzić nowe fakty z zadanych lub wcześniej wyprowadzonych faktów nie przechowywanych w bazach danych. Podstawowym problemem organizacji procesu poszukiwania odpowiedzi na zadane pytanie jest uniknięcie pełnego przeszukiwania przestrzeni wszystkich możliwych wniosków. Jest to typowe podejście w badaniach nad systemami sztucznej inteligencji w "rozumieniu" tekstu, z uwzględnieniem składniowych i semantycznych struktur danych* [3], [62], [44].

W monografii tej termin *wyszukiwanie informacji* będziemy używali w znaczeniu drugim, tj. jako systemy przechowywania i wyszukiwania informacji.

W systemach wyszukiwania informacji możemy wyróżnić cztery podstawowe problemy [79]:

(a) *Analiza tekstu.* Zajmuje się ona tworzeniem charakterystyk wyszukiwawczych dokumentów, tj. automatycznym indeksowaniem (wyborem odpowiednich deskryptorów) [78], [101], [108], automatycznym streszczeniem i rozumieniem języka naturalnego [4]. Dwa główne kierunki badań opierają się na statystycznych lub lingwistycznych metodach analizy tekstu. W podejściu statystycznym [92] korzysta się z dodatkowych informacji zawartych w słownikach i tezaurusach, a w podejściu lingwistycznym z metod reprezentacji wiedzy w lingwistycznych systemach [5].

(b) *Struktury informacyjne.* Prowadzone w tym zakresie badania są ukierunkowane na opracowanie takich metod organizacji wyszukiwania dokumentów, które zapewnią skrócenie czasu wyszukiwania i pozwolą na przechowywanie zbioru dokumentów w możliwie jak najmniejszym obszarze pamięci komputera [30].

(c) *Analiza kwerend.* Główne zagadnienia dotyczą [38]:

- dokładnego wyrażania potrzeb informacyjnych,
- sposobu przedstawiania struktur informacyjnych,
- środków dostępnych w celu łączenia deskryptorów,
- związków deskryptorów dostrzeganych zarówno przez ekspertów, jak i przypadkowych użytkowników, które umożliwiają wyszukiwanie relewantnych i odrzucanie nirelewantnych dokumentów.

(d) *Mechanizmy wyszukiwania informacji.* Główne kierunki badań związane są z metodami wyszukiwania zbiorów opisów dokumentów odpowiadających potrzebie informacyjnej wyrażonej w kwerendzie, tj. określenie sposobu jak ma być wyszukany zbiór, aby dostarczyć najlepsze wyniki i jakie operacje mają być wykonane na zbiorach dokumentów będących odpowiedziami na kwerendy składające się z pojedynczych deskryptorów. Nie mogą występować sprzeczności pomiędzy łączeniem częściowych odpowiedzi zgodnie z matematyczną teorią wyszukiwania a intuicyjnymi wyobrażeniami użytkownika formułującego swoje kwerendy jako kombinacje deskryptorów. Problem jest bardziej złożony, kiedy wprowadzają się wagi deskryptorów w celu wyposażenia użytkowników w środki do wyrażania ważności deskryptorów w kwerendzie i umożliwienia szeregowania dokumentów zgodnie ze stopniem relewancji.

Gwałtowny rozwój technologii produkcji sprzętu komputerowego w ostatnim dziesięcioleciu umożliwił szybki rozwój sieci komputerowych. Zainteresowanie użytkowników tymi środkami informatyki przyczyniło się do powstania koncepcji rozproszonego przetwarzania danych. Wydaje się, że przyszłość informatyki należy do systemów rozproszonych realizowanych w sieciach teleprzetwarzania. Sieci komputerowe umożliwią większym grupom użytkowników korzystanie z istniejących systemów wyszukiwania informacji. Sieć komputerowa pozwala na nawiązywanie łączności i komunikowanie się pomiędzy różnymi komputerami. Z praktyki wiadomo, że większość lokalnych systemów wyszukiwania informacji i zbiorów danych była tworzona niezależnie od siebie. Powstawały systemy różnorodne pod względem języków informacyjno-wyszukiwawczych i metod wyszukiwania informacji.

Pojawia się problem, gdy chcemy połączyć wiele różnych niezależnych systemów wyszukiwania informacji w jeden duży system wyszukiwania informacji, zwany systemem rozproszonym, użytkowany w sieci komputerowej. W systemach tego typu powstaje wiele nowych problemów dotyczących, np.: metod wymiany informacji między poszczególnymi podsystemami, różnorodności węzłów, ułatwienia użytkownikom dostępu do lokalnych zbiorów dokumentów, prywatności, integralności i niezawodności transmisji danych.

W pracach dotyczących sieci komputerowych bada się głównie aspekt komunikacyjny między poszczególnymi komputerami, nie są natomiast głębiej analizowane mechanizmy uzyskiwania odpowiedzi w takich sieciach. Gdy nie zna się szczegółowo tych mechanizmów, może się zdarzyć, że wyszukane odpowiedzi są bądź nieprawidłowe, bądź jeśli są prawidłowe, można je otrzymać znacznie prostszą drogą [68].

W przypadku rozproszonych systemów pojawia się nowa grupa "globalnych" użytkowników mających dostęp do wszystkich "lokalnych"

podsystemów wyszukiwania informacji, np. "globalny" użytkownik rozproszonego systemu bibliotecznego może mieć dostęp do wszystkich podsystemów. Szukając określonej książki, użytkownik będzie miał możliwość sprowadzenia jej z dowolnej biblioteki, jeśli w tej, w której aktualnie szuka, jej nie ma.

Cel i wyniki pracy

Większość dotychczasowych systemów wyszukiwania informacji i zbiorów danych była tworzona niezależnie od siebie. Chcąc połączyć wiele różnych niezależnych systemów wyszukiwania informacji w jeden duży, rozproszony system wykorzystywany w sieci komputerowej, należy rozwiązać wiele nowych problemów nie występujących w tradycyjnych scentralizowanych systemach wyszukiwania informacji.

Celem niniejszej monografii jest przeanalizowanie zagadnienia połączenia „lokalnych” systemów wyszukiwania informacji w jeden rozproszony system. W pracy rozpatrzono zasadniczy problem dotyczący określenia struktury logicznej mechanizmu uzyskiwania odpowiedzi w rozproszonym systemie opartym na globalnym teaurusie z wagami.

W większości dotychczasowych automatycznych systemów wyszukiwania informacji stosowano statyczną przestrzeń dokumentów, w tym sensie, że dokumentom odpowiadały stałe pozycje w przestrzeni. Statyczną przestrzeń dokumentów indukują systemy klasyfikacji, które nie zawsze odpowiadają aktualnym potrzebom użytkowników. Aby móc zaspokajać wymagania użytkowników, zaproponowano użycie w rozproszonym systemie dynamicznej przestrzeni dokumentów, w której dokumenty mogą zmieniać swoje pozycje w przestrzeni zgodnie z potrzebami użytkowników. Aktualizacja wag deskryptorów opisów dokumentów na podstawie subiektywnej opinii użytkowników o trafności odpowiedzi pozwala utworzyć opisy lepiej odzwierciedlające ich rzeczywistą treść. W pracy zaproponowano metody modyfikacji opisów dokumentów na podstawie ocen użytkowników o relewantności dokumentów należących do odpowiedzi. Pokazano, że ze zmodyfikowanej przestrzeni dokumentów korzyści czerpią nie tylko ci użytkownicy, którzy brali udział w modyfikowaniu dokumentów. Nowi użytkownicy korzystają z "doświadczeń" zebranych podczas poprzednich poszukiwań. Podano również statystyczne kryterium zmniejszenia redundancji w strukturach wyszukiwawczych oraz oszacowano pesymistyczne koszty kwerend w rozproszonym systemie.

W rozdziale 1 dokonano krótkiego przeglądu najważniejszych problemów teorii systemów wyszukiwania informacji. W rozdziale 2 zdefiniowano system wyszukiwania informacji oparty na teaurusie z

wagami, w którym na zbiorze deskryptorów określona jest relacja uogólnienia, a wagi deskryptorów występują w opisach dokumentów i w kwerendach. W rozdziale 3 na bazie takich systemów zbudowano model jednorodnego rozproszonego systemu wyszukiwania informacji. Podano formalny jego opis i udowodniono zachowanie w rozproszonym systemie podstawowej cechy inkluzyjności systemów opartych na tezaurusach. Określono operacje na odpowiedziach z lokalnych podsystemów w celu przygotowania globalnej odpowiedzi. W rozdziale 4 zaproponowano modyfikację klasycznej metody list inwersyjnych uwzględniającej relację uogólnienia i wagi deskryptorów w celu ich wykorzystania w rozproszonym systemie. Metodę tę można traktować jako logiczną implementację rozproszonego systemu. W rozdziale 5 przedstawiono własności rozproszonych systemów wyszukiwania informacji będących szczególnym przypadkiem rozproszonego systemu zdefiniowanego w rozdziałach 2 i 3. Sposób włączenia tych systemów do heterogenicznego rozproszonego systemu podano w rozdziale 6. Geometryczną interpretację rozproszonego systemu podano w rozdziale 7.

Aktualizacja wag deskryptorów użytych do opisu dokumentu na podstawie subiektywnej opinii użytkowników o trafności odpowiedzi pozwala utworzyć opisy lepiej odzwierciedlające rzeczywistą treść dokumentów. W rozdziale 8 zaproponowano metody modyfikacji opisów dokumentów na podstawie ocen użytkowników o relewantności dokumentów należących do odpowiedzi na zadaną kwerendę. W rozdziale 9 podano statystyczne kryterium wyboru wartości progowych dla list inwersyjnych w lokalnych podsystemach, z których dokumenty rzadko używane można usunąć, aby zmniejszyć redundancję dokumentów w rozproszonym systemie. W rozdziale 10 oszacowano pesymistyczne koszty kwerend uwzględniające koszty obliczeń w węzłach i koszty przesyłania danych między węzłami.

Rozproszony system wyszukiwania informacji został zaimplementowany na mikrokomputerze IBM PC/AT, na którym zasymulowano pracę takiego systemu. W dodatku monografii, w tabelach, przedstawiono współczynniki podobieństw odpowiedzi i list inwersyjnych przed i po adaptacji rozproszonego systemu.

1. PRZEGLĄD ZAGADNIEN W TEORII WYSZUKIWANIA INFORMACJI

1.1. Klasyfikacja i indeksowanie dokumentów

Wyszukiwanie informacji jest to proces odszukiwania w dowolnym zbiorze tych dokumentów, które dotyczą wskazanego w kwerendzie tematu (przedmiotu) lub zawierają fakty, informacje. Dokumentem nazywamy dowolny, zapisany na jakimkolwiek materialnym nośniku, tekst wyrażający myśl ludzką, stanowiący logiczną całość i zawierający informacje o jego źródle i/lub twórcy. Dokumentem jest więc nie tylko książka, artykuł, opis patentowy lub świadectwo autorskie itp., lecz także oddzielny fragment tego tekstu, np. paragraf, rozdział, akapit itd. A oto kilka przykładów dokumentów:

(a) W tradycyjnych systemach wyszukiwania informacji obiektami wyszukiwania są bibliograficzne opisy dokumentów bądź też same dokumenty lub ich kopie [87]. Dlatego też te systemy nazywane są systemami wyszukiwania dokumentów.

(b) W systemach informacyjnych przedsiębiorstw dokumentami mogą być różnego rodzaju zarządzenia i pisma urzędowe przechowywane w komputerach osobistych [22].

(c) W instytucjach naukowych dokumentami mogą być także raporty i publikacje udostępniane użytkownikom poprzez lokalne sieci komputerowe.

(d) W rozległych sieciach komputerowych dokumenty mogą być rozumiane jako rozproszone komunikaty i informacje do różnych użytkowników, system wyszukiwania informacji wybiera z węzłów te komunikaty, które dotyczą danego użytkownika.

(e) Ponadto dokumentami mogą być różnego rodzaju teksty, wzory struktur chemicznych, rysunki itp. wykorzystywane w systemach, które umożliwiają szybkie odszukanie np. reakcji chemicznych określonego typu, sposobów otrzymywania związków chemicznych określonej klasy i reakcji, w które wchodzi dany związek chemiczny oraz informacji o związkach i reakcjach chemicznych według warunków ich przeprowadzania.

Każdy dokument, przy wprowadzaniu do systemu wyszukiwania informacji, podlega indeksowaniu, tzn. procesowi tworzenia dokładnego i w pełni informacyjnego opisu dokumentu. Deskryptory służące do opisów dokumentów

mogą być fragmentami tekstów (słowa lub zdania), fragmentami wzorów chemicznych, rysunków itp.

Najprostszą techniką indeksowania jest metoda polegająca na dołączaniu do każdego opisu dokumentu listy, tzw. terminów indeksowych [30]. W procesie wyszukiwania system wyznacza zazwyczaj wartość funkcji podobieństwa pomiędzy zbiorami terminów kwerendy a zbiorami terminów indeksowych przypisanym dokumentom i na jej podstawie decyduje o tym, czy dany dokument ma być wyszukany czy też nie. Wybór odpowiedniej procedury indeksowania determinuje więc możliwości wyszukiwawcze systemu [18], [22], [32], [100].

Najistotniejszymi rodzajami informacji podlegającymi klasyfikacji są [30]:

- (i) słowa (terminy, deskryptory),
- (ii) dokumenty stanowiące bazę systemu.

Klasyfikacja słów ma za zadanie grupowanie ich w klasy (np. synonimów), które umożliwiają bardziej adekwatne porównywanie kwerendy z dokumentami opisanymi tymi słowami. Natomiast klasyfikacja dokumentów ma na celu zwiększenie szybkości procesu wyszukiwania tylko do pewnych podzbiorów dokumentów. Istnieje ścisły związek między klasyfikacją słów a klasyfikacją dokumentów, bowiem na podstawie słów wyznaczonych do opisu dokonuje się klasyfikacji dokumentów. Stosowanie dobrej klasyfikacji słów i dokumentów zapewnia wysoką kompletność i dokładność odpowiedzi [22], [32], [77], [100], [101].

1.2. Modele systemów wyszukiwania informacji

Aby osiągnąć wspomniane wyżej cele, opracowano wiele rozmaitych metod i algorytmów wyszukiwania. Opierają się one na różnych matematycznych modelach i często wymagają odpowiednich organizacji danych. Scharakteryzujemy teraz najważniejsze modele systemów. Dobre systemy wyszukiwania informacji to takie, których czas oczekiwania na odpowiedzi jest krótki. Warunek ten wymaga odpowiedniej organizacji zbiorów. Handlowe systemy oparte są najczęściej na organizacji list inwersyjnych [86], chociaż istnieją także inne organizacje zbiorów gwarantujące efektywny dostęp do dokumentów [81]. Organizacja list inwersyjnych pozwala na bezpośredni dostęp do dokumentów zawierających odpowiednie deskryptory w swoich opisach. Operacje *sumy*, *przekroju* i *dopełnienia* list inwersyjnych odpowiadają boolowskim operatorom dysjunkcji *OR*, koniunkcji *AND* i negacji *NOT* deskryptorów występujących w kwerendzie. Ostatni z operatorów jest trudny do zrealizowania w

praktyce, gdyż odpowiada mu negacja zbioru, która oznacza branie całej przestrzeni z wyjątkiem negowanego zbioru. W związku z tym wprowadza się często, zamiast operatora *NOT*, operator dwuargumentowy *BUT NOT*, który ma takie same znaczenie co *NOT*, lecz zaprzeczenie dotyczy jedynie przestrzeni wyznaczonej przez pierwszy argument. Jednak dla wielu zastosowań takie rozwiązanie jest niewystarczające i dlatego wprowadza się dodatkowe operatory kontekstowe, np. *ADJ*, *HITH*, *SAME* itp. Metoda list inwersyjnych po pewnych modyfikacjach umożliwia interpretację kwerend z tak rozszerzonym zbiorem operatorów. Takie rozszerzenie przyjęto np. w systemie STAIRS [37]. Metoda list inwersyjnych umożliwia także łatwe utrzymanie hierarchii określonej na zbiorze deskryptorów [30].

Jedną z wad tradycyjnego wyszukiwania boolowskiego jest fakt, że nie jest możliwe określenie stopnia relewancji, tzn. czy dany dokument jest bardziej relewantny od drugiego. Jest bardzo pożądane, aby dokumenty w odpowiedzi były uszeregowane zgodnie z oceną ich relewancji [64]. W tym celu wymaga się, aby dokumenty nie były oceniane tylko jako relewantne lub nirelewantne do zadanej kwerendy, lecz także żąda się podania stopnia relewancji. Są trzy czynniki wpływające na sposoby szeregowania dokumentów [63]:

- (a) wagi deskryptorów w opisach dokumentów,
- (b) wagi deskryptorów w kwerendach,
- (c) miary podobieństwa pomiędzy kwerendami a opisami.

Opisy różnych dokumentów zawierających te same deskryptory nie są rozróżnialne w modelach niewagowych, natomiast wprowadzenie wag do opisów dokumentów umożliwia precyzyjniejsze opisywanie dokumentów bez konieczności rozbudowy tezausa. Wagi deskryptorów określają stopień ważności poszczególnych deskryptorów w kwerendach i w opisach dokumentów. Korzyści z wprowadzenia wag do opisów dokumentów przemawiają za stosowaniem w praktyce modeli wagowych, jednak należy wspomnieć o dodatkowych aspektach związanych m.in. z indeksowaniem dokumentów i przechowywaniem informacji. Przypisywanie wag deskryptorom można dokonywać ręcznie, opierając się na częstotliwościach wystąpień deskryptorów w treści dokumentu (wskaźniki roli) lub możliwościach deskryptorów odróżniania jednego dokumentu od drugiego (wskaźnik więzi) [86]. Jednym z najlepiej znanych sposobów indeksowania dokumentów opartych na rozkładzie częstotliwości deskryptorów jest model Poissona, zaproponowany przez Booksteina i Swansona [11], [12] i rozwinięty przez Hartera [34].

Przyjmuje się, że terminy nie będące deskryptorami (nie związane z treścią dokumentu) są równo rozmieszczone wśród dokumentów systemu wyszukiwania informacji. Prawdopodobieństwo wystąpienia danego deskryptora pewną liczbę razy w określonym dokumencie dane jest rozkładem Poissona. W dwupoissonowym modelu indeksowania dokumentów zakłada się, że

poszczególne deskryptory występują częściej w dokumentach relewantnych niż nirelewantnych, przez co otrzymujemy dwa różne rozkłady prawdopodobieństw dla relewantnych i nirelewantnych dokumentów. Wagi w opisach dokumentów są wyliczane na podstawie parametrów rozkładu Poissona.

Wagi deskryptorów w kwerendach wyrażają wymaganą ważność pojęć w opisach dokumentów. Istnieje wiele różnych interpretacji wag deskryptorów [23]. Tradycyjne boolowskie systemy wyszukiwania informacji można traktować jako szczególny przypadek wyszukiwania "ważonego". Wystąpienie deskryptora w opisie dokumentu lub w kwerendzie jest równoważne jednostkowej wadze dla tego deskryptora, a brak deskryptora odpowiada zerowej wadze deskryptora. Ponadto wymaga się, aby kwerendy z deskryptorami były kierowane do systemów, w których opisy dokumentów składają się tylko z samych deskryptorów i odpowiednio kwerendy zawierające pary deskryptorów z wagami były kierowane do systemów, w których opisy dokumentów zawierają deskryptory z wagami. Żąda się, aby operatory użyte w kwerendach nie miały wpływu na interpretację wag. Warunek ten gwarantuje, że odpowiedzi na częściowe kwerendy składające się z pojedynczych deskryptorów można łączyć zgodnie z operatorami, nie bacząc na wielkości wag deskryptorów w kwerendach. Użytkownik może łączyć odpowiedzi na różne kwerendy i otrzymywać taki sam wynik jak w przypadku odpowiedzi na modyfikowane kwerendy.

Klasyczna teoria zbiorów jest wykorzystywana w tradycyjnych "nieważonych" systemach, natomiast posługując się teorią zbiorów rozmytych [109], można znacznie lepiej zbadać i opisać własności systemów wagowych. W tym przypadku waga deskryptora jest funkcją charakterystyczną zbioru rozmytego deskryptorów opisu dokumentu. Podstawowe założenie nie ogranicza się do tego, czy występuje lub nie występuje dany deskryptor w opisie dokumentu, lecz do jego częściowej przynależności do dokumentu. Również dokumenty mogą mieć częściową przynależność do odpowiedzi.

Rozmytym systemem wyszukiwania informacji jest czwórka (D, T, Q, f) , gdzie: D jest zbiorem opisów dokumentów, Q zbiorem kwerend, $f: Q \times D \rightarrow [0, 1]$ funkcją zgodności przyporządkowującą każdej parze (q, d) , gdzie $q \in Q$ i $d \in D$, liczbę $f(q, d)$ z przedziału $[0, 1]$ oznaczającą stopień członkostwa kwerendy q w opisie dokumentu d [70], [90]. Jeśli funkcja f przyjmuje wartości $\{0, 1\}$, to otrzymujemy tradycyjny niewagowy model boolowski. W modelu rozmytym funkcja przynależności dla koniunkcji, alternatywy i negacji kwerend jest zdefiniowana następująco:

$$f(q_1 \wedge q_2, d) = \min(f(q_1, d), f(q_2, d)), \quad f(q_1 \vee q_2, d) = \max(f(q_1, d), f(q_2, d)),$$

$$f(\neg q, d) = 1 - f(q, d).$$

Dla ograniczającego operatora *AND* małe wagi deskryptorów w kwerendach znaczą "małe ograniczenia" i dlatego podczas wykonywania

operacji *AND* można zaliczać do odpowiedzi wszystkie dokumenty zawierające deskryptory – operandy *AND*, natomiast dla dopuszczalnego operatora *OR* małe wagi znaczą "trochę dopuszczalne" i podczas wykonywania operacji *OR* można nie wyszukiwać tych dokumentów. Problemy interpretacji i kombinacji boolowskich operacji *AND*, *OR*, *NOT* w kwerendach z ważonymi deskryptorami rozwiązano w pracach [9], [16].

Alternatywne rozwiązanie podano w [16]. Wagi w kwerendach interpretuje się nie jako wagi relewancji, ale jako wartości progowe. Opisy dokumentów dla danych wag w kwerendach są oceniane pod kątem osiągania wartości progowych. Opisy dokumentów otrzymują wysokie współczynniki korelacji, jeśli wagi deskryptorów są równe wartościom progowym. Współczynnik korelacji pomniejsza się, jeśli zwiększają się różnice wag deskryptorów, opisów dokumentów względem wartości progowych. Rozważa się problem jak duże wartości progowe mogą wystąpić w kwerendach, a później sprawdza się, ile tych wartości było przekroczonych. Ponieważ wartości progowe stosowane są niezależnie do poszczególnych deskryptorów, więc wszystkie składowe relewancji mogą być łączone zwykłą funkcją *minimum* lub *maksimum*.

Najważniejszym problemem w modelu progowym jest interpretacja wag w kwerendach. Niskie wagi deskryptorów z kwerend w modelu rozmytym oznaczają, że obecność lub nieobecność tych deskryptorów w opisie dokumentu nie jest taka ważna, natomiast niskie wagi progowe w kwerendach wymagają, aby poszczególne deskryptory w opisach dokumentów miały również niskie wagi. Na pierwszy rzut oka taka interpretacja wag wydaje się przekonywująca. Jednak wadą jest niewyszukiwanie dokumentów, które w swoich opisach zawierają deskryptory nie występujące w kwerendzie i w następstwie tego opisy te mogą być rzadko wyszukiwane.

W tradycyjnych modelach w trakcie szeregowania dokumentów nie uwzględnia się boolowskich operatorów. Dokumenty należące do odpowiedzi mogą być uszeregowane zgodnie z liczbą wspólnych deskryptorów z opisu i kwerendy. W takim modelu opisy dokumentów mogą być przedstawione w postaci binarnych wektorów, których składowe wskazują na wystąpienia poszczególnych deskryptorów. Kwerendy muszą mieć dokładnie taką samą postać jak opisy dokumentów. Deskryptory rozpinają deskryptorową przestrzeń wektorów, w której opisy dokumentów i kwerendy są punktami tej przestrzeni [80]. Pierwszym krokiem w wyszukiwaniu dokumentów w takiej przestrzeni wektorowej jest porównanie wszystkich współczynników korelacji wektorów opisów dokumentów z wektorem kwerendy. Binarną przestrzeń wektorową można łatwo uogólnić na ciągłą przestrzeń, w której osie przedstawiające deskryptory przyjmują wartości rzeczywiste wag deskryptorów opisów dokumentów i kwerend. Wagi mogą mieć takie same znaczenie jak np. w modelu rozmytym.

Obliczanie współczynników korelacji wektorów nie powoduje takich trudności w procesie wyszukiwania jakie powstają przy realizacji operatorów boolowskich. Jeśli kwerendę traktować będziemy jako punkt w przestrzeni wektorowej, to do odpowiedzi powinny należeć te opisy, które leżą najbliżej kwerendy. Z drugiej strony, współczynnik korelacji wektorów preferują te deskryptory, których rzuty na wektor kwerendy są największe. Opisy dokumentów zawierające dużą liczbę deskryptorów mają większą szansę wyszukania, ponieważ ich korelacja nakładania z kwerendą uwzględniająca liczbę wspólnych deskryptorów będzie najprawdopodobniej większa.

Najczęściej stosowaną miarą podobieństwa jest korelacja kosinusowa [81] normalizująca stopień podobieństwa. Matematycznie miara ta przedstawia kosinus kąta pomiędzy wektorem opisu dokumentu a wektorem kwerendy. Opisy dokumentów należące do odpowiedzi są szeregowane zgodnie z wartością kąta pomiędzy wektorami opisu i kwerendy. Norault, McGill i Koll zbadali 24 różne miary podobieństwa pomiędzy opisami dokumentów i kwerendami [63].

Duży wpływ na wartości współczynników korelacji mają pojedyncze deskryptory w kwerendach i opisach dokumentów. Deskryptory występujące jednocześnie w kwerendzie i opisie dokumentu zwiększają wartość współczynnika korelacji, wystąpienie deskryptora tylko w jednym z wektorów pomniejsza wartość współczynnika, ponieważ znaczy to, że kwerenda i opis mniej odpowiadają sobie, natomiast brak danego deskryptora w obu wektorach nie ma żadnego wpływu na wartość współczynnika.

Możliwe jest uwzględnianie związków deskryptorów pochodzących ze sprzężenia zwrotnego [72]. Jeśli użytkownik oceni wyszukany dokument jako nierelevantny, to uważa się, że istnieją negatywne związki pomiędzy deskryptorami w kwerendzie i w opisie dokumentu. Przeciwnie, jeśli relevantny dokument nie jest wyszukany, to uważa się, że istnieją pozytywne związki pomiędzy nie dobranymi deskryptorami kwerendy i tego opisu. Aby wykorzystać te związki w procesie wyszukiwania, najpierw oblicza się pewne współczynniki podobieństwa dla par deskryptorów, a następnie dodaje się do miary stopnia podobieństwa wartość funkcji obliczonej dla pozytywnej korelacji i pomniejsza się o wartość funkcji dla negatywnej korelacji $\mu(d,q) = \cos(d,q) + f(\text{pozytywne związki deskryptorów}) - f(\text{negatywne związki deskryptorów})$. Ponieważ $\cos$ jest miarą znormalizowaną, funkcja f też musi być znormalizowana. Niestety, Raghavan i Yu nie wskazali jak otrzymywać informacje o relewantności dokumentów nie wyszukanych.

Standardowy model systemu wyszukiwania boolowskiego można łatwo rozszerzyć, nakładając mniejsze ograniczenie na operator **AND** oraz stawiając większe wymaganie operatorowi **OR** [84]. Do obliczania współ-

czynnika korelacji kwerendy do opisu dokumentu w n -wymiarowej przestrzeni deskryptorowej L_p ($1 \leq p \leq \infty$) stosuje się miarę p -podobieństwa określoną następująco:

(a) $sim_p^{OR}(q, d) = [q^p \cdot d^p]^{1/p} / |q|$, dla kwerend q będących alternatywą ważonych deskryptorów,

(b) $sim_p^{AND}(q, d) = 1 - [q^p \cdot (1-d)^p]^{1/p} / |q|$, dla kwerend q będących koniunkcją ważonych deskryptorów, gdzie d^p i q^p oznaczają odpowiednio wektory opisu dokumentu d i kwerendy q , których elementy mają postać potęg odpowiednio d_i^p i q_i^p , 1 jest wektorem, którego elementami są 1 , „ $\cdot$ ” oznacza iloczyn skalarny wektorów, $|q| = (\sum_{i=1}^n q_i^p)^{1/p}$ jest normą wektora kwerendy q w n -wymiarowej przestrzeni deskryptorowej L_p .

Dla $p=1$ oba współczynniki stają się iloczynami skalarnymi znormalizowanymi przez rozmiar kwerendy (liczbę ważonych deskryptorów w kwerendzie). Dla $p=\infty$, współczynnik sim_p^{OR} jest funkcją *maksimum* znormalizowaną przez maksymalną wagę deskryptora z kwerendy, natomiast sim_p^{AND} jest funkcją *minimum* znormalizowaną przez minimalną wagę deskryptora z kwerendy. Gdy $p=\infty$ i wagi deskryptorów w kwerendach i opisach dokumentów są ograniczone do wartości $0, 1$, to otrzymamy standardowy model boolowski. W ten sposób można przejść od wyszukiwania rozmytego do modelu przestrzeni wektorowej. Najlepsze wyniki osiąga się, gdy $2 \leq p \leq 5$ [85]. Należy jednak podkreślić, że dla dużych p pojawiają się pewne niekonsekwencje w wyszukiwaniu rozmytym, natomiast dla małych wartości p współczynnik korelacji w modelu przestrzeni wektorowej powinien być raczej normalizowany przez długość wektora dokumentu, a nie przez długość wektora kwerendy. Z teoretycznego punktu widzenia model ten ma duże znaczenie.

Pierwsze prace dotyczące wykorzystania przestrzeni wektorowych w modelach systemów wyszukiwania informacji zawierały wiele niekonsekwencji. Najczęściej zakładano, że wektory odpowiadające deskryptorom są parami ortogonalne. Takie założenie jest nierealistyczne. Mimo że przeprowadzano różne próby łagodzące takie założenia, operacje na wektorach w tych przestrzeniach nie były zgodne z podstawowymi regułami i aksjomatami algebraicznymi [71].

Użycie modelu przestrzeni wektorowej wymaga dokładnego określenia kilku podstawowych aspektów, m.in. interpretacji wartości w opisach dokumentów, wymiaru przestrzeni, definicji wektorów bazowych, korelacji pomiędzy wektorami deskryptorów (jeśli nie są one ortonormalne) ważnych podczas definiowania odpowiedniej reprezentacji kwerend i opisów dokumentów. Proponując nową metodę reprezentacji wektorów deskryptorów w terminach odpowiednio wybranych ortonormalnych wektorów bazowych, możemy

zdefiniować nowy uogólniony model przestrzeni wektorowej [102]. Każdemu wektorowi bazowemu przestrzeni liniowej można jednoznacznie przyporządkować iloczyn kanoniczny składający się z n deskryptorów. Reprezentacja wektorów deskryptorów może być explicite przedstawiona w 2^n -wymiarowym układzie kartezjańskim. Dowolna boolowska kwerenda może być transformowana na wektor w przestrzeni 2^n -wymiarowej. Wektory opisów dokumentów, wyrażone jako liniowe kombinacje wektorów bazowych, mogą być uszeregowane w odpowiedzi tak, jak w konwencjonalnym modelu przestrzeni wektorowej. Miarą podobieństwa może być iloczyn skalarny wektorów opisu dokumentu i kwerendy. Nie ma konieczności wprowadzania specjalnych funkcji jako miary podobieństwa. Model ten może być traktowany jako zunifikowany model systemu wyszukiwania informacji, ponieważ ma wszystkie zalety uogólnionych modeli boolowskich i przestrzeni wektorowych. Wyniki eksperymentów [106] dowodzą, że model ten jest efektywniejszy niż standardowe implementacje modeli przestrzeni wektorowych.

W przestrzeni wektorowej dokumenty położone blisko siebie tworzą grupę dokumentów o podobnych opisach. Taka grupa może być reprezentowana przez jeden opis. Ta idea może być wykorzystana do tworzenia grup powiązanych dokumentów i wyszukiwania jedynie w ramach pewnych grup. Przykładem takiego podejścia może być następująca metoda [88]:

1. Porównując pewne parametry wybranego dokumentu z odpowiednimi parametrami pozostałych dokumentów zbioru, wybiera się i grupuje te dokumenty, których parametry są dostatecznie podobne.

2. Dla każdej utworzonej grupy wybiera się pewien element reprezentatywny, zwany wektorem centroidalnym. Jest on później uważany za reprezentanta całego zbioru dokumentów tej grupy.

3. Wyszukiwanie przebiega w dwóch etapach: zgłoszoną kwerendę najpierw porównuje się z centroidami wszystkich grup dokumentów; w drugim przeszukiwaniu porównuje się kwerendę ze wszystkimi dokumentami z grup o dużym stopniu podobieństwa z centroidem.

Dwustopniowe wyszukiwanie można rozszerzyć do trzystopniowego, lub nawet wyższego stopnia, polegającego na grupowaniu wektorów centroidalnych w szersze grupy, a następnie łączeniu tych grup w jeszcze szersze itd. W tym przypadku pierwsze przeszukiwanie odbywa się w ramach centroidów grup najwyższego poziomu; w ten sposób z przeszukiwania na danym poziomie otrzymujemy kilka centroidów reprezentujących grupy niższego poziomu itd., aż do wyznaczenia paru grup dokumentów, które muszą być przeszukane osobno.

Wydajność takiego wielostopniowego wyszukiwania jest różna w zależności od zastosowanego procesu wiązania i typu rozpatrywanego zbioru: jest ona większa, jeśli zbiór można podzielić na rozłączne grupy mniej więcej jednakowej wielkości, natomiast zmniejsza się, jeśli grupy

"nakładają się" i wielkość ich różni się bardzo od wspólnej wartości idealnej. Oczywiście, wyszukiwanie wielostopniowe będzie nieopłacalne, gdy interesujących użytkownika dokumentów nie ma w tych grupach, które mają być przeszukiwane osobno, bowiem znajdujące się poza nimi dokumenty relewantne nie mogą być odszukane.

Oprócz modeli rozmytych i modeli przestrzeni wektorowej prowadzi się badania nad probabilistycznymi modelami indeksowania i wyszukiwania informacji. Oznaczmy przez:

- A klasę wszystkich żądań użytkowników,
- C klasę wszystkich dokumentów w systemie,
- $B \subseteq A$ klasę podobnych żądań (deskrytory z kwerend należą do określonego podzbioru deskrytorów),
- $D \subseteq C$ klasę podobnych dokumentów (opisanych przez deskrytory należące do określonego podzbioru deskrytorów),
- $b_k \in B$ pojedyncze żądanie,
- $d_m \in D$ pojedynczy dokument,
- $R \subseteq A \times C$ relację relewancji, $(b_k, d_m) \in R$, jeśli d_m był oceniony jako relewantny do b_k .

W modelach probabilistycznych zakłada się, że prawdopodobieństwa dla par (żądanie, dokument) są niezależne od wyboru pary. Istnieje kilka podstawowych modeli systemów z różnymi sposobami interpretacji i obliczania prawdopodobieństwa relewancji. Obliczanie poszczególnych prawdopodobieństw staje się głównym problemem. W Modelu-1 prawdopodobieństwo relewancji dokumentu $P(R|B, d_m)$ jest obliczane względem klasy żądań i do jego obliczania zwykle wykorzystuje się częstotliwości wystąpień deskrytorów w dokumencie.

Model-1 jest wykorzystywany do indeksowania dokumentów. Probabilistyczne modele indeksowania próbują maksymalizować prawdopodobieństwo dla każdej informacji zawartej w opisie dokumentu ocenionego przez użytkownika jako relewantny. Przykładem probabilistycznego indeksowania może być wspomniany wcześniej dwupoissonowy model Hartera. W probabilistycznym modelu Marona i Kuna [47], a później rozszerzonym przez Coopera i Marona [20], grupuje się użytkowników w celu obliczenia prawdopodobieństwa relewancji danego dokumentu.

W Modelu-2 prawdopodobieństwo relewancji $P(R|b_k, D)$ klasy podobnych dokumentów jest obliczane względem żądania użytkownika i do obliczenia używa się sprzężenia zwrotnego. W modelu Roberstona i Spark Jones [75], który dokładniej został zbadany przez van Rijsbergena [98], dokumenty grupuje się w celu obliczenia prawdopodobieństwa relewancji dla danego użytkownika.

W Modelu-0 prawdopodobieństwo relewancji $P(R|B, D)$ oblicza się pomiędzy klasami dokumentów i żądań, do jego obliczenia używa się

współczynników korelacji pomiędzy kwerendami a opisami dokumentów, np. liczby wspólnych deskryptorów.

W Modelu-3 oblicza się prawdopodobieństwo relewancji pojedynczego dokumentu dla danego użytkownika. W zunifikowanym modelu probabilistycznym [74] pojedyncze zdarzenie "użytkownik dokonujący oceny relewancji pojedynczego dokumentu" jest traktowane jako przecięcie dwóch zbiorów zdarzeń: "pojedynczy użytkownik z klasą dokumentów" i "grupa użytkowników z pojedynczym dokumentem". Prawdopodobieństwo relewancji dane jest wzorem

$$P(R|b_k, d_m) = \frac{P(R|B, p_m) \cdot P(R|b_k, D)}{P(R|B, D)}$$

Ważny jest fakt, że w zunifikowanym modelu dozwolone jest wystąpienie tylko jednego deskryptora w żądaniu (potrzebie informacyjnej) oraz w opisie dokumentu. Dla dowolnej liczby deskryptorów problem ten został rozwiązany [6] przez założenie niezależności deskryptorów w opisie dokumentu.

Można pokazać także istnienie związków pomiędzy modelem probabilistycznym i modelem przestrzeni wektorowej [6]. Jeśli dwupoissonowy model jest użyty do indeksowania, to funkcja oceny jest równa współczynnikowi korelacji wektorów w przestrzeni wektorowej, w której wagi deskryptorów są równe częstotliwościom wystąpień deskryptorów w treści dokumentu. Wagi deskryptorów kwerend przyjmują postać logarytmu ilorazu pomiędzy oczekiwanymi częstotliwościami deskryptorów w relewantnych i nierelewantnych dokumentach, tj. $q_i = \log(u_i/v_i)$, gdzie q_i jest wagą deskryptora kwerendy, a u_i , v_i są parametrami dwupoissonowego rozkładu.

W modelach probabilistycznych utrudnieniem jest zbyt duża liczba deskryptorów. Obliczenie wszystkich prawdopodobieństw może przekraczać zarówno możliwości komputera, jak i wyczerpywać cierpliwość użytkownika. Van Rijsbergen [98] zaproponował, aby w kwerendach występowały tylko deskryptory. W jego opinii same deskryptory dają dobre oszacowanie prawdopodobieństwa.

Istnieją również metody integracji boolowskich i probabilistycznych systemów, w których boolowskie kwerendy są interpretowane jako sposób określania zależności deskryptorów w relewantnych zbiorach dokumentów [21]. W tych systemach nie można używać operatora *NOT*.

Dostępność na rynku tanich mini- i mikrokomputerów o wysokich parametrach użytkowych sprzyja rozwojowi systemów rozproszonego przetwarzania danych. Wiele koncepcji rozproszonego przetwarzania danych wymusza pewien rodzaj przetwarzania w sieciach komputerowych i decentralizację trzech jego elementów: przetwarzania, bazy danych i sterowania siecią komputerową. We wszystkich znanych formach przetwarzania danych istnieje wiele możliwych rozwiązań, zależnie od stopnia zdecen-

tralizowania każdego z tych elementów [53], [65].

W pracach dotyczących sieci komputerowych bada się głównie aspekt komunikacyjny, nie są natomiast głębiej analizowane mechanizmy uzyskiwania odpowiedzi w takich sieciach. Gdy nie zna się szczegółowo tych mechanizmów, może się zdarzyć, że wyszukane odpowiedzi są bądź nieprawidłowe, bądź jeśli są prawidłowe, można je otrzymać znacznie prostszą drogą [68]. Po raz pierwszy zagadnienia sposobu znajdowania odpowiedzi w rozproszonych systemach na podstawie uzyskanych odpowiedzi częściowych z lokalnych systemów rozważano w pracy [66], w której mówi się o rozproszeniu systemu z punktu widzenia:

- (a) języka,
- (b) obiektów,
- (c) pamięci.

W pierwszym wypadku mamy do czynienia z faktem "podziału" języka systemu między użytkowników, tj. udostępnienia każdemu lokalnemu użytkownikowi fragmentu globalnego języka zgodnie z jego potrzebami. W drugim przypadku są "podzielone" między użytkowników obiekty rozproszonego systemu, tzn. każdy użytkownik ma dostęp do informacji nie o wszystkich dokumentach, lecz tylko o tych, które są mu "przypisane". W ostatnim przypadku mamy do czynienia z "podziałem" pamięci między użytkowników, z których każdy może mieć swój podsystem. Może też istnieć tylko jeden centralny system, z którego korzystają wszyscy użytkownicy z odpowiednimi ograniczeniami.

W systemie Pawlaka dokumenty charakteryzuje się poprzez ich własności [67]. W tym celu wprowadza się pojęcia atrybutu i wartości atrybutu. Podobny problem, dotyczący głównie organizacji maszyn baz danych, był rozważany w [46].

Własności rozproszonych systemów zbudowanych z lokalnych systemów opartych na modelach wagowych i hierarchicznych zostały przedstawione w pracach [48], [49], [56]. Pokazano również zachowanie własności inkluzyjności w rozproszonym systemie, pozwalającej na tworzenie łańcuchów odpowiedzi na łańcuchy kwerend, tj. na poszukiwanie odpowiedzi na "ściślejszą" kwerendę wśród odpowiedzi uzyskanych na "mniej precyzyjną" kwerendę. Klasyczne listy inwersyjne, istniejące w lokalnych podsystemach wyszukiwania informacji opartych na różnych modelach, mogą być wykorzystane w rozproszonych systemach [48], [49], [51]. Są one wykorzystywane do utworzenia globalnej odpowiedzi bez konieczności tworzenia globalnych struktur informacyjnych.

Pierwszy prototypowy rozproszony system wyszukiwania informacji [45] umożliwia symulację działania rozproszonego systemu. Wykorzystywany jest on również do określania zasad projektowania i implementacji rzeczywistych rozproszonych systemów użytkowych.

2. MATEMATYCZNY MODEL SYSTEMU WYSZUKIWANIA INFORMACJI

2.1. Opis lokalnego systemu wyszukiwania informacji

W każdym lokalnym systemie S_j tezaurus T_j jest skończonym, niepustym zbiorem terminów t spełniających następujące warunki:

- (i) istnieje niepusty zbiór $T_{j0} \subset T$, nazwany zbiorem deskryptorów,
- (ii) określona jest relacja równoważności $\phi_j \subset T_j \times T_j$ taka, że żadne dwa różne deskryptory nie pozostają w relacji ϕ_j oraz dla każdego elementu zbioru $T_j \setminus T_{j0}$ istnieje element zbioru deskryptorów, pozostający z nim w relacji synonimii, tj.

$$(\forall X \in T_j / \phi_j) (\text{card}(X \cap T_{j0}) = 1).$$

Relację ϕ_j nazywamy relacją synonimii, a terminy t_1, t_2 będące w tej relacji — terminami synonimicznymi,

(iii) określona jest przeciwzrotna, przechodnia, a więc i przeciw-symetryczna relacja $R_j \subset T_{j0} \times T_{j0}$, którą nazywamy relacją uogólnienia. Jeśli dwa deskryptory t_1, t_2 są w relacji $t_1 R_j t_2$, to mówimy, że deskryptor t_1 jest ogólniejszy niż deskryptor t_2 albo że deskryptor t_2 jest bardziej szczegółowy niż t_1 . Zakładamy ponadto, że spełniony jest warunek

$$\neg(t_1 R_j t_2 \vee t_2 R_j t_1) \wedge t_1 R_j t_3 \wedge t_2 R_j t_4 \Rightarrow t_3 \neq t_4,$$

tzn. dla dwóch różnych deskryptorów t_1, t_2 nie będących w relacji R_j nie istnieje szczegółowy deskryptor t_3 taki, że $t_1 R_j t_3 \wedge t_2 R_j t_3$.

Elementy ze zbioru $T_j \setminus T_{j0}$, mające znaczenie pomocnicze, nazywamy askryptorami. Z podanej definicji tezaury wynika, że dla każdego askryptora istnieje dokładnie jeden synonimiczny deskryptor.

Zbiór deskryptorów T_{j0} z relacją uogólnienia R_j może być przedstawiony w postaci zorientowanego grafu acyklicznego przy następujących założeniach:

- (i) wierzchołkami grafu G są elementy ze zbioru deskryptorów T_{j0} ,
- (ii) każda krawędź (t_1, t_2) grafu G odpowiada relacji $t_1 R_j t_2$; wierzchołek t_1 nazywamy początkiem krawędzi, a wierzchołek t_2 jej końcem,

(iii) z przechodniości relacji R_j ($t_1 R_j t_2 \wedge t_2 R_j t_3 \Rightarrow t_1 R_j t_3$) oraz z poprzedniego założenia wynika, że krawędź (t_1, t_3) też powinna być elementem grafu G . Dla uproszczenia postaci grafu G nie będziemy rozważać krawędzi typu (t_1, t_3) .

Zorientowany graf acykliczny z jednym korzeniem, jedyną drogą prowadzącą od korzenia do danego wierzchołka grafu G i jedyną krawędzią wchodzącą do każdego wierzchołka nazywamy drzewem zorientowanym. Tezaurus T jest grafem zorientowanym składającym się z rozłącznych drzew zorientowanych.

W wyniku klasyfikacji powstaje opis dokumentu d zdefiniowany jako zbiór par (deskryptor, waga)

$$t(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$$

spełniający warunki:

$$(i) \quad (\forall 1 \leq i \leq k) \quad (0 < w_i \leq 1), \quad (1)$$

$$(ii) \quad \sum_{i=1}^k w_i \leq 1, \quad (2)$$

$$(iii) \quad t_1 \neq t_2 \wedge t_1 R_j t_2 \Rightarrow (t_1, w_1) \notin t(d) \vee (t_2, w_2) \notin t(d). \quad (3)$$

Niech

$$t(d_1) = \{(t_{11}, w_{11}), (t_{12}, w_{12}), \dots, (t_{1k_1}, w_{1k_1})\} \quad (4)$$

$$t(d_2) = \{(t_{21}, w_{21}), (t_{22}, w_{22}), \dots, (t_{2k_2}, w_{2k_2})\} \quad (5)$$

będą opisami dokumentów d_1 i d_2 .

Powiemy, że opis $t(d_1)$ zawiera się w opisie innego dokumentu $t(d_2)$ z dokładnością c ($0 \leq c \leq 1$), jeśli

$$(i) \quad \forall_{1 \leq i \leq k_1} \quad \exists_{1 \leq m \leq k_2} \quad t_{1i} = t_{2m} \vee t_{1i} R_j t_{2m} \quad (6)$$

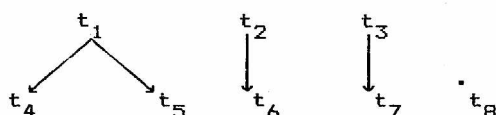
$$(ii) \quad \forall_{1 \leq i \leq k_1} \quad c w_{1i} \leq \sum_{k \in N_1} w_{2k} \quad (7)$$

gdzie $N_1 = \{m: (t_{1i} = t_{2m} \vee t_{1i} R_j t_{2m}) \wedge 1 \leq m \leq k_2\}$.

Jeśli opisy $t(d_1)$ i $t(d_2)$ spełniają warunki (6)–(7), to będziemy pisać $t(d_1) \succ_{jc} t(d_2)$.

Przykład 1

Niech $T_j = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$ będzie tezauresem, w którym



Opisy dokumentów d_1 i d_2 są postaci

$$t(d_1) = \{(t_1, 0.3), (t_2, 0.4), (t_3, 0.3)\}$$

$$t(d_2) = \{(t_4, 0.1), (t_5, 0.1), (t_6, 0.3), (t_7, 0.2), (t_8, 0.3)\}$$

Relacja $t(d_1) \succ_{0.5} t(d_2)$ jest spełniona, ponieważ każda nierówność w poniższym układzie jest prawdziwa

$$\begin{cases} 0.3 * 0.5 \leq 0.1 + 0.1 \\ 0.4 * 0.5 \leq 0.3 \\ 0.3 * 0.5 \leq 0.2 \end{cases}$$

natomiast relacja $t(d_1) \succ_{0.7} t(d_2)$ nie jest spełniona, ponieważ dwie poniższe nierówności są fałszywe

$$\begin{cases} 0.3 * 0.7 \leq 0.1 + 0.1 & (\text{fałsz}) \\ 0.4 * 0.7 \leq 0.3 & (\text{fałsz}) \\ 0.3 * 0.7 \leq 0.2 & (\text{fałsz}) \end{cases}$$

Kwerendą nazywamy skończony ciąg deskryptorów z sumą wag równą jedności, nie zawierający powtórzeń i uogólnień występujących deskryptorów. Kwerenda ma więc taką samą postać jak opis dokumentu z warunkami (1)–(3), w którym suma wag jest równa jedności. W podsystemie S_j będziemy używać wartości $\mu(q, t)$ oznaczającej stopień zawierania się kwerendy q w opisie dokumentu $t(d)$.

Lokalnym systemem wyszukiwania informacji opartym na tezaurysie z wagami nazwiemy czwórkę $S_j = (T_j, D_j, Q_j, f_j)$, gdzie T_j jest tezaurysie, D_j zbiorem dokumentów, Q_j zbiorem kwerend, f_j odwzorowaniem $Q_j \times [0, 1] \rightarrow D_j \times [0, 1]$ przyporządkowującym parze (kwerenda q , żądana dokładność odpowiedzi c) zbiór par postaci (d, v) , gdzie element d należy do podzbioru zbioru dokumentów, a v jest stopniem zawierania się kwerendy q w opisie dokumentu d . Odpowiedzią podsystemu S_j na kwerendę $q \in Q_j$ nazywać będziemy $f_j(q, c) \subset D_j \times [0, 1]$.

2.2. Podobieństwo kwerend i opisów dokumentów

Twierdzenie 1

Niech $t(d_1)$ i $t(d_2)$ będą opisami dokumentów postaci (4) i (5).

Jeśli $t(d_1) \succ_{c_1} t(d_2) \wedge c_1 \geq c_2$, to $t(d_1) \succ_{c_2} t(d_2)$.

Dowód

Ponieważ $c_1 \geq c_2 \implies w_{1i}c_1 \geq w_{1i}c_2$, stąd dla każdego ($1 \leq i \leq k_1$) nierówność

$w_{1i}c_2 \leq \sum_{k \in N_i} w_{2k}$ w wyrażeniu (7) jest prawdziwa.

■

Z wyrażeń (6) i (7) wynika, że relacja $\succ_{jc}$ jest zwrotna i antysymetryczna. Pokażemy teraz, że relacja $\succ_{jc}$ jest quasi-przechodnia.

Przyjmijmy, że

- opisy dokumentów d_1 i d_2 są postaci (4) i (5),
- $t(d_3) = \{(t_{31}, w_{31}), (t_{32}, w_{32}), \dots, (t_{3k_3}, w_{3k_3})\}$ jest opisem d_3 ,
- $N_i^{12} = \{m: (t_{1i} = t_{2m} \vee t_{1i} R_j t_{2m}) \wedge 1 \leq m \leq k_2\}$, dla $1 \leq i \leq k_1$,
- $N_i^{23} = \{m: (t_{2i} = t_{3m} \vee t_{2i} R_j t_{3m}) \wedge 1 \leq m \leq k_3\}$, dla $1 \leq i \leq k_2$,
- $N_i^{13} = \{m: (t_{1i} = t_{3m} \vee t_{1i} R_j t_{3m}) \wedge 1 \leq m \leq k_3\}$, dla $1 \leq i \leq k_1$.

W celu zwiększenia czytelności wzorów w dalszej części tego podrozdziału, pisząc o własnościach lokalnych podsystemów S_j , będziemy omijali wskaźnik j we wszystkich wyrażeniach.

Twierdzenie 2

Jeśli $t(d_1) \succ_{c_1} t(d_2) \wedge t(d_2) \succ_{c_2} t(d_3)$, to $t(d_1) \succ_c t(d_3) \wedge c = c_1 c_2$.

Dowód

Z przechodniości relacji uogólnienia R wynika, że

$$(\forall (t_{1i}, w_{1i}) \in t(d_1)) (\exists (t_{3i}, w_{3i}) \in t(d_3)) (t_{1i} R t_{3i})$$

Z nierówności

$$c_1 w_{1i} \leq \sum_{k \in N_i^{12}} w_{2k} \quad \text{ i } \quad c_2 w_{2i} \leq \sum_{k \in N_i^{23}} w_{3k}$$

otrzymujemy, że

$$w_{1i} c_1 \leq \sum_{k \in N_i^{12}} w_{2k} \leq \sum_{k \in N_i^{12}} \left(\frac{1}{c_2} \sum_{k \in N_i^{23}} w_{3k} \right) = \frac{1}{c_2} \sum_{k \in N_i^{13}} w_{3k}.$$

Ostatnia równość wynika z zależności $N_i^{13} = \bigcup_{k \in N_i^{12}} N_i^{23}$.

Pokazaliśmy, że $(\forall (t_{1i}, w_{1i}) \in t(d_1)) (w_{1i} c_1 c_2 \leq \sum_{k \in N_i^{13}} w_{3k})$.

■

Z twierdzenia 2 wynika, że jeśli $t(d_1) \succ_{c_1} t(d_2) \succ_{c_2} t(d_3)$, to dokładność w relacji $t(d_1) \succ_c t(d_3)$ jest nie mniejsza niż iloczyn $c_1 c_2$. Relacja zawierania się $\succ_c$ jest zdefiniowana dla kwerend i par (opis, kwerenda) oraz (kwerenda, opis) tak samo jak dla opisów. Jeśli relacja zawierania $\succ_c$ zachodzi pomiędzy q i $t(d)$, to powiemy, że kwerenda q zawiera się w opisie dokumentu $t(d)$ z dokładnością c .

Dla opisów $t(d_1), t(d_2)$ postaci (4), (5) spełniających warunek (6) możemy zdefiniować stopień zawierania się opisu $t(d_1)$ w opisie $t(d_2)$

$$\mu(t(d_1), t(d_2)) = \sum_{i=1}^{k_1} \min \left(w_{1i}, \sum_{k \in N_i} w_{2k} \right), \quad (8)$$

gdzie $N_i = \{m: (t_{1i} = t_{2m} \vee t_{1i} R_j t_{2m}) \wedge 1 \leq m \leq k_2\}$, dla $1 \leq i \leq k_1$.

Przykład 2

Dla opisów dokumentów d_1 i d_2 z przykładu 1 stopień zawierania się opisu $t(d_1)$ w opisie $t(d_2)$ wynosi

$$\mu(t(d_1), t(d_2)) = \min(0.3, 0.1 + 0.1) + \min(0.4, 0.3) + \min(0.3, 0.2) = 0.7.$$

■

Twierdzenie 3

Stopień zawierania się opisów dokumentów spełnia warunek

$$0 \leq \mu(t(d_1), t(d_2)) \leq 1.$$

Dowód

Korzystając z nierówności $\min\left(w_{1i}, \sum_{k \in N_i} w_{2k}\right) \leq w_{1i}$

i wyrażenia (8), otrzymujemy

$$\mu(t(d_1), t(d_2)) = \sum_{i=1}^{k_1} \min\left(w_{1i}, \sum_{k \in N_i} w_{2k}\right) \leq \sum_{i=1}^{k_1} w_{1i} \leq 1.$$

Ponieważ wagi deskryptorów w opisach dokumentów $t(d_1)$ i $t(d_2)$ są nieujemne, stąd $\mu(t(d_1), t(d_2)) \geq 0$.

■

Stopień zawierania się μ zdefiniowany jest dla kwerend i par (kwerenda, opis) tak samo jak dla opisów. Zależność pomiędzy dokładnością c relacji $\succ_c$ a stopniem zawierania się μ podamy w następnym twierdzeniu.

Twierdzenie 4

Niech opisy dokumentów $t(d_1)$, $t(d_2)$ będą postaci (4), (5) oraz

$$\sum_{i=1}^{k_1} w_{1i} = 1.$$

(i) Jeśli dla każdego $(t_{1i}, w_{1i}) \in t(d_1)$

$$w_{1i} c = \sum_{k \in N_i} w_{2k}, \text{ to } \mu(t(d_1), t(d_2)) = c.$$

(ii) $\forall_{0 \leq c \leq 1} t(d_1) \succ_c t(d_2) \Rightarrow \mu(t(d_1), t(d_2)) \geq c.$

(iii) $\mu(t(d_1), t(d_2)) = c_1 \Rightarrow \exists_{0 < c \leq c_1} t(d_1) \succ_c t(d_2).$

Dowód

(i) Korzystając z wyrażeń (7), (8), otrzymujemy

$$\mu(t(d_1), t(d_2)) = \sum_{i=1}^{k_1} \min \left(w_{1i}, \sum_{k \in N_i} w_{2k} \right) = \sum_{i=1}^{k_1} \min \left(w_{1i}, w_{1i} c \right) = \sum_{i=1}^{k_1} w_{1i} c = c.$$

(ii) Z dwóch poniższych nierówności

$$w_{1i} c \leq w_{1i} \quad \text{ i } \quad w_{1i} c \leq \sum_{k \in N_i} w_{2k}$$

otrzymujemy, że $w_{1i} c \leq \min \left(w_{1i}, \sum_{k \in N_i} w_{2k} \right)$, stąd

$$c = \sum_{i=1}^{k_1} w_{1i} c \leq \sum_{i=1}^{k_1} \min \left(w_{1i}, \sum_{k \in N_i} w_{2k} \right) = \mu(t(d_1), t(d_2)).$$

(iii) Oznaczmy przez c

$$c = \min_{1 \leq i \leq k_1} \left\{ \sum_{k \in N_i} w_{2k} / w_{1i} \right\} \quad (9)$$

(c spełnia następujący warunek: $0 < c \leq 1$).

Najpierw pokażemy, że $c_1 \geq c$. Korzystając z wyrażeń (7), (8), otrzymujemy

$$\begin{aligned} c_1 = \mu(t(d_1), t(d_2)) &= \sum_{i=1}^{k_1} \min \left(w_{1i}, \frac{w_{1i}}{w_{1i}} \sum_{k \in N_i} w_{2k} \right) \geq \\ &\geq \sum_{i=1}^{k_1} \min \left(w_{1i}, w_{1i} \min_{1 \leq i \leq k_1} \left\{ \sum_{k \in N_i} w_{2k} / w_{1i} \right\} \right) = \sum_{i=1}^{k_1} \min(w_{1i}, w_{1i} c) = c. \end{aligned}$$

Teraz pokażemy, że opis dokumentu $t(d_1)$ jest zawarty w opisie dokumentu $t(d_2)$ z dokładnością c . Z wyrażenia (9) otrzymujemy nierówność

$$\forall_{1 \leq i \leq k_1} \quad c \leq \sum_{k \in N_i} w_{2k} / w_{1i}, \quad \text{która jest zgodna z (7).}$$

2.3. Własności odpowiedzi lokalnego systemu

Odpowiedzią systemu na kwerendę $q \in Q$ z wymaganą dokładnością c nazywać będziemy

$$f(q, c) = \{ (d, v) : d \in D \wedge q \succ_c t(d) \wedge v = \mu(q, t(d)) \}. \quad (10)$$

Relacja zawierania $\succ_c$ pozwala na wyszukiwanie dokumentów w zbiorze opisów dokumentów spełniających warunek: dla każdego deskryptora z kwerendy iloczyn jego wagi i dokładności c jest mniejszy od sumy wag deskryptorów z opisu dokumentu, ściślej: $\mu(q, t(d))$ określa stopień zawierania się kwerendy q w opisie dokumentu $t(d)$ należącego do odpowiedzi.

Podstawową własność odpowiedzi systemu, a mianowicie zależność zachodzącą pomiędzy dokładnością c a stopniem zawierania się kwerendy w opisach dokumentów należących do odpowiedzi, można wyrazić w postaci następującego twierdzenia.

Twierdzenie 5

Jeśli $(d, v) \in f(q, c)$, to $v \geq c$.

Dowód wynika bezpośrednio z twierdzenia 4(ii). ■

Twierdzenie odwrotne nie jest prawdziwe, tzn. jeśli $v \geq c$, to nie zawsze $(d, v) \in f(q, c)$.

Przykład 3

Niech $q = \{ (t_1, 0.6), (t_2, 0.4) \}$ i $t(d) = \{ (t_1, 0.7), (t_2, 0.01), (t_3, 0.29) \}$. Dokument d o opisie $t(d)$ będzie należał do odpowiedzi wtedy, gdy wymagana dokładność c odpowiedzi będzie mniejsza lub równa 0.025, pomimo że stopień zawierania $v = \mu(q, t(d)) = \min(0.6 + 0.7) + \min(0.4, 0.01) = 0.61$.

Twierdzenie 6

Jeśli $c_1 \geq c_2$, to $f(q, c_1) \subset f(q, c_2)$.

Dowód

Niech $(d, v) \in f(q, c_1)$, stąd $q \succ_{c_1} t(d)$. Korzystając z twierdzenia 1 otrzymujemy, że $q \succ_{c_2} t(d)$, tzn. $(d, v) \in f(q, c_2)$.

Inkluzyjnością systemu nazywamy cechę pozwalającą na tworzenie łańcuchów odpowiedzi na łańcuchy kwerend, tj. na poszukiwanie odpowiedzi na "ściślejszą" kwerendę wśród odpowiedzi uzyskanych na "mniej precyzyjną" kwerendę. W rozpatrywanym systemie warunek inkluzyjności systemu można sformułować w następującej postaci

$$\{q_1, q_2\} \subset Q \wedge q_1 \succ_c q_2 \wedge c_1 \leq c_2 \Rightarrow f(q_1, c_1) \supset f(q_2, c_2). \quad (11)$$

Dowód związku (11) podano w [58]. Własność inkluzyjności systemu ma duże znaczenie praktyczne, pozwala ona ograniczyć poszukiwanie odpowiedzi na coraz precyzyjniej formułowane kwerendy do odpowiedzi kwerend mniej precyzyjnych. Użytkownik w trakcie konwersacji może precyzować swoje kwerendy bądź to przez zastępowanie deskryptorów ogólniejszych przez deskryptory ściślejsze, bądź to przez dodawanie nowych deskryptorów, bądź też przez zwiększenie dokładności odpowiedzi. Analizę innych własności podsystemu związanych z odpowiedziami na kwerendy postaci $q_1 + q_2$ i $q_1 \cdot q_2$ można znaleźć w pracy [58].

2.4. Organizacja lokalnych list inwersyjnych

Główną częścią każdego rzeczywistego systemu przeznaczonego do przechowywania i wyszukiwania informacji jest zorganizowany w odpowiedni sposób zbiór wyszukiwawczy. Jedną z najczęściej implementowanych w praktyce organizacji zbiorów jest inwersyjna organizacja zbioru wyszukiwawczego. Inwersyjny zbiór wyszukiwawczy składa z list inwersyjnych. Ze względu na tradycję przez pojęcie "lista inwersyjna" będziemy rozumieć zbiór par (dokument, waga).

Niech $t(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ będzie opisem dokumentu $d \in D$. Dla każdego deskryptora $t_i \in T_0$ lista inwersyjna $L(t_i)$ ma postać

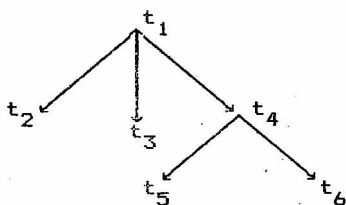
$$\forall_{t_i \in T_0} L(t_i) = \{(d, v) : (t_i, w_i) \in t(d) \wedge v = w_i\}. \quad (12)$$

Waga dokumentu d jest równa wadze z jaką deskryptor t wystąpił w opisie

dokumentu $t(d)$. Liczba list inwersyjnych jest równa mocy zbioru deskryptorów T_0 . Dokument, którego opis zawiera m deskryptorów, będzie należał do m list inwersyjnych. W dalszych rozważaniach, ze względów praktycznych, dokument utożsamiać będziemy z jego numerem (adresem).

Przykład 4

Niech zbiór deskryptorów T_0 ma postać $T_0 = \{t_1, t_2, t_3, t_4, t_5, t_6\}$. Relacje uogólnienia R zachodzące pomiędzy deskryptorami przedstawiono na rys. 1 w postaci grafu:



Rys. 1. Graf opisujący relację uogólnienia pomiędzy deskryptorami
Fig. 1. Graph describing generalization relation for descriptors

Do opisów dokumentów postaci:

$$t(d_1) = \{(t_2, 0.4), (t_3, 0.3), (t_5, 0.3)\}$$

$$t(d_2) = \{(t_3, 0.7), (t_4, 0.3)\}$$

$$t(d_3) = \{(t_3, 0.4), (t_4, 0.6)\}$$

$$t(d_4) = \{(t_2, 0.1), (t_3, 0.4), (t_4, 0.5)\}$$

$$t(d_5) = \{(t_5, 0.4), (t_6, 0.6)\}$$

listy inwersyjne przyjmą postać:

$$L(t_1) = \emptyset$$

$$L(t_2) = \{(d_1, 0.4), (d_4, 0.1)\}$$

$$L(t_3) = \{(d_1, 0.3), (d_2, 0.7), (d_3, 0.4), (d_4, 0.4)\}$$

$$L(t_4) = \{(d_2, 0.3), (d_3, 0.6), (d_4, 0.5)\}$$

$$L(t_5) = \{(d_1, 0.3), (d_5, 0.4)\}$$

$$L(t_6) = \{(d_5, 0.6)\}.$$

Na listach inwersyjnych $L(t)$ określone są działania sumy $\bar{\cup}$ i przekroju $\cap$. Sumy i iloczyny list inwersyjnych są listami inwersyjnymi.

Definicja 1

Niech $(d_1, v_1) \in L(t_1)$ i $(d_2, v_2) \in L(t_2)$.

$$(i) \quad (d_1, v_1) = (d_2, v_2) \Leftrightarrow (d_1 = d_2) \wedge (v_1 = v_2)$$

$$(ii) \quad (d, v) \in L(t_1) \bar{\cup} L(t_2) \Leftrightarrow \exists_{d_1 \neq d_2} \left[(d, v) = (d_1, v_1) \vee (d, v) = (d_2, v_2) \right] \vee \exists_{d_1 = d_2} \left[(d, v) = (d_1, v_1 + v_2) \right].$$

$$(iii) \quad (d, v) \in L(t_1) \cap L(t_2) \Leftrightarrow \forall_{d_1 = d_2} (d, v) = (d_1, v_1 + v_2).$$

Jeśli dany dokument należy do kilku różnych list inwersyjnych, to waga dokumentu należącego do sumy lub przekroju tych list jest sumą wag jakie dokument miał w poszczególnych listach.

Przykład 5

Opisy dokumentów i postać list inwersyjnych jest taka sama jak w przykładzie 4.

Wynikiem sumy $\bar{\cup}$ list $L(t_2)$ i $L(t_4)$ jest lista inwersyjna postaci:

$$L(t_2) \bar{\cup} L(t_4) = \{(d_1, 0.4), (d_2, 0.3), (d_3, 0.6), (d_4, 0.6)\}$$

Przecięcie $\cap$ list $L(t_2)$ i $L(t_3)$ ma postać:

$$L(t_2) \cap L(t_3) = \{(d_1, 0.7), (d_4, 0.5)\}.$$

Twierdzenie 7

Niech $t_1, t_2 \in T_0$.

$$\forall_{t_1 R t_2} L(t_1) \cap L(t_2) = \emptyset.$$

Dowód

Założmy nie wprost, że $\exists (d, v) \in L(t_1) \cap L(t_2)$.

Z definicji 1(iii) wynika, że

$$\exists_d (t_1, v_1) \in t(d) \wedge \exists_d (t_2, v_2) \in t(d),$$

co daje sprzeczność z warunkiem (2). ■

Definicja listy inwersyjnej nie uwzględnia relacji uogólnienia na zbiorze deskryptorów.

Definicja 2

Oznaczamy przez X_i zbiór wszystkich deskryptorów ściślejszych od deskryptora t_i .

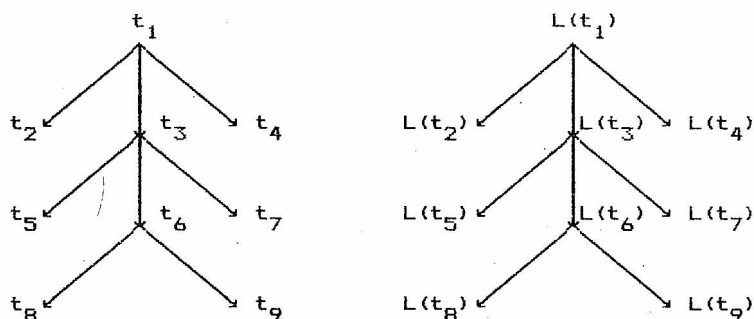
$$\forall_{t_i \in T_0} X_i = \{t: t \in T_0 \wedge t_i = t \vee t_i R t\}.$$

Lista inwersyjna $LR(t_i)$ dla deskryptora t_i uwzględniająca relację uogólnienia R ma postać:

$$LR(t_i) = \bigcup_{t \in X_i} L(t).$$

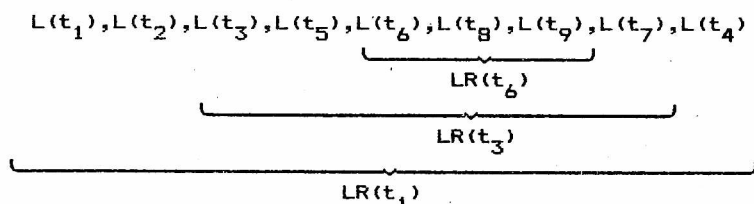
■

Tezaurus T ma strukturę drzew, których węzłami są deskryptory. Możemy również zbudować podobne drzewa dla list inwersyjnych [51], w których węzłami będą listy $L(t)$ (rys. 2).



Rys. 2. Drzewa tezaurusu i list inwersyjnych
Fig. 2. Thesaurus and inverted files trees

Te drzewa mogą być przekształcone w drzewa binarne. Celowe jest zapisanie list w pamięci komputera w liniowym porządku, zgodnie z metodą prefix przechodzenia binarnego drzewa



W praktyce nie trzeba budować list inwersyjnych $LR(t_i)$, suma $LR(t_i) = \bigcup_{t \in X_i} L(t)$ zawiera dokumenty należące do kolejnych list $L(t)$. Ze względu na relację uogólnienia R , przechowywanie wszystkich list postaci $LR(t)$ znacznie zwiększyłoby redundancję danych.

Przykład 6

Niech zbiór deskryptorów i zbiór opisów dokumentów będą takie same jak w przykładzie 4. Dla deskryptora t_4 lista inwersyjna uwzględniająca relację uogólnienia R ma postać:

$$LR(t_4) = L(t_4) \cup L(t_5) \cup L(t_6) = \{(d_1, 0.3), (d_2, 0.3), (d_3, 0.6), (d_4, 0.5), (d_5, 1.0)\}.$$

■

Obliczanie sumy $\bigcup_{t \in X_i} L(t)$, zgodnie z definicją 1(ii), pozwala na usunięcie powtórzeń dokumentów. W trakcie usuwania powtórzeń sumuje się wagi dokumentów występujących wielokrotnie. Obliczone w ten sposób wagi dokumentów z listy $LR(t_i)$ są sumami wag dokumentów, w opisach których znajdują się deskryptory ściślejsze od deskryptora t_i . W czasie pracy systemu wyszukiwania informacji tak obliczona suma list inwersyjnych pozwala na bardzo proste obliczenie stopnia podobieństwa kwerendy q do opisu dokumentu $t(d)$, za pomocą miary (8), bez żadnych dodatkowych operacji arytmetycznych. Wagi dokumentów znajdujące się na listach inwersyjnych $LR(t_i)$ będą składnikami sumy (8).

Na zbiorze $LR(t)$ określona jest relacja porządku liniowego $\geq$, szeregująca elementy zbioru $LR(t)$ ze względu na wielkość wag, tj.

$$\begin{aligned} & \forall (d_i, v_i) \in LR(t) \\ & (d_k, v_k) \in LR(t) \quad (d_i, v_i) \geq (d_k, v_k) \Leftrightarrow v_i \geq v_k. \end{aligned}$$

W dalszych rozważaniach zakładamy, że elementy (d_i, v_i) w listach inwersyjnych $LR(t)$ są umieszczone w kolejności malejących wag dokumentów. Łatwo zauważyć, że pomiędzy zbiorami $LR(t_i)$ i $\bigcup_{t \in X_i} L(t)$ zachodzi następujący związek

$$\text{card } LR(t_i) \leq \sum_{t \in X_i} \text{card } L(t).$$

Podczas obliczania sumy $\bar{U}$ list inwersyjnych wszystkie powtórzenia dokumentów są usuwane.

Definicja 3

(i) Lista $L(t_1)$ zawiera się w liście $L(t_2)$. Piszemy $L(t_1) \subset L(t_2)$ wtedy i tylko wtedy, gdy

$$\forall (d_1, v_1) \in L(t_1) \quad \exists (d_2, v_2) \in L(t_2) \quad d_1 = d_2 \wedge v_1 \geq v_2.$$

(ii) Lista $L(t_1)$ zawiera się w liście $L(t_2)$ z dokładnością b ($0 \leq b \leq 1$). Piszemy $L(t_1) \subset_b L(t_2)$ wtedy i tylko wtedy, gdy

$$\exists 0 \leq b \leq 1 \quad \forall (d_1, v_1) \in L(t_1) \quad \exists (d_2, v_2) \in L(t_2) \quad d_1 = d_2 \wedge v_1 \geq v_2 b. \quad (13)$$

Gdy $b=1$, definicja "zawierania się z dokładnością b " pokrywa się z definicją "zawierania się", natomiast gdy $b=0$, "zawieranie się" zbiorów jest równoważne z zawieraniem zbiorów w teorii mnogości.

Z definicji "zawierania" i definicji list $LR(t)$ wynika prosta inkluzja

$$\forall t \in T_0 \quad L(t) \subset LR(t).$$

Twierdzenie 8

Niech $a = \min\{v_k/v_i : (d, v_k) \in L(t_1) \wedge (d, v_i) \in L(t_2)\}$.

(i) Jeśli $L(t_1) \subset_a L(t_2)$, to $\forall_{0 \leq b \leq a} L(t_1) \subset_b L(t_2)$.

(ii) $\forall_{t_2 R t_1} \exists_{0 \leq b \leq a} LR(t_1) \subset_b LR(t_2)$.

Dowód

(i) Musimy sprawdzić prawdziwość warunku (13) w definicji 3(ii) dotyczącego wag dokumentów. Ponieważ a zostało określone jako minimum ilorazów wag dokumentu d należącego do list $LR(t_1)$ i $LR(t_2)$ dla wszystkich dokumentów z listy $L(t_1)$, stąd dla każdej wagi dokumentu d należącego do listy $L(t_1)$ jest spełniony warunek $v_k \geq v_i a$. Ponieważ dla każdego $0 \leq b \leq a$ nierówność $v_k \geq v_i a \geq v_i b$ jest prawdziwa, więc $L(t_1) \subset_b L(t_2)$.

(ii) Niech $X_1 = \{t : t \in T_0 \wedge (t_1 = t \vee t_1 R t)\}$

$X_2 = \{t : t \in T_0 \wedge (t_2 = t \vee t_2 R t)\}$

będą zbiorami deskryptorów bardziej szczegółowych względem deskryptorów odpowiednio t_1 i t_2 .

Z implikacji $\forall_{t_1, t_2} t_2 R t_1 \Rightarrow X_1 \subset X_2$ wynika, że

$$LR(t_1) = \bigcup_{t \in X_1} L(t) \subset \bigcup_{t \in X_2} L(t) = LR(t_2).$$

Dalsza część dowodu wynika z twierdzenia 8(i).

Z twierdzenia 8(ii) wynika następująca własność

$$\forall_{t_1 R t_2} LR(t_1) \cap LR(t_2) \neq \emptyset.$$

Definicja 4

Listę $LR(t) \big|_a$ nazywać będziemy listą obciętą przez wartość a .
Do listy $LR(t) \big|_a$ będą należeć pary (dokument, waga) z listy $LR(t)$,
których wagi są nie mniejsze od liczby a , tj.

$$LR(t) \big|_a = \{ (d, v) : (d, v) \in LR(t) \wedge v \geq a \}.$$

■

Katwo zauważyć, że dla $a=0$ prawdziwa jest równość $LR(t) \big|_a = LR(t)$
oraz dla $0 \leq a \leq 1$ zachodzi $LR(t) \big|_a \subset LR(t)$. Podstawowe własności list
obciętych można przedstawić w postaci twierdzenia.

Twierdzenie 9

- (i) $LR(t_1) \subset LR(t_2) \Rightarrow LR(t_1) \big|_a \subset LR(t_2) \big|_a$.
- (ii) Jeśli $a_1 \geq a_2$, to $LR(t) \big|_{a_1} \subset LR(t) \big|_{a_2}$.
- (iii) $\forall_{t_1, t_2} t_2 Rt_1 \wedge LR(t_1) \big|_{a_1} \subset_b LR(t_2) \big|_{a_2} \Leftrightarrow \exists_{0 \leq b \leq 1} a_1 b \leq a_2$.

Dowód związków (i)–(ii) wynika bezpośrednio z definicji 3(i) oraz
definicji 4.

Dowód (iii)

Jeśli $t_2 Rt_1$, to na podstawie twierdzenia 8(ii) istnieje takie b , że
 $LR(t_1) \subset_b LR(t_2)$. Z definicji 3(ii) otrzymujemy warunek dla wag dokumentu
należącego do obu list, tj. $v_1 b \leq v_2$. Stąd wynika, że a_1, a_2 muszą
spełniać podobną zależność, tj. $a_1 b \leq a_2$.

■

W dalszych rozważaniach będziemy zakładać, że liczba a ma postać
 $a=cw$, gdzie w jest wagą deskryptora z kwerendy odwołującej się do danej
listy inwersyjnej, c jest dokładnością odpowiedzi wymaganą przez
użytkownika (znaczenie c jest takie same jak w wyrażeniu (7)).

W następnej definicji podamy sposób obliczania iloczynu $\cap$ list
inwersyjnych obciętych przez $a=cw$.

Definicja 5

Niech $(d_i, v_i) \in LR(t_1) \mid_{cw_1}$ i $(d_k, v_k) \in LR(t_2) \mid_{cw_2}$.

$$(d, v) \in LR(t_1) \mid_{cw_1} \cap LR(t_2) \mid_{cw_2} \Leftrightarrow$$

$$\Leftrightarrow \forall (d, v) = (d_i, \min(v_i, w_1) + \min(v_k, w_2)) \wedge v_i \geq cw_1 \wedge v_k \geq cw_2, \\ d_i = d_k$$

■

Łatwo można pokazać istnienie zależności $w \geq c$ zachodzącej pomiędzy v i c . Ponieważ $v_i \geq cw_1$ i $v_k \geq cw_2$, więc $v = \min(v_i, w_1) + \min(v_k, w_2) \geq \min(cw_1, w_1) + \min(cw_2, w_2) = c(w_1 + w_2) = c$.

Definicja 5 jest uogólnieniem definicji 1(iii). Jeśli w definicji 5 przyjmujemy $c=0$ i $w_1=w_2=1$, to otrzymamy definicję równoważną z definicją 1(iii).

Definicja 6 (odpowiedź na kwerendę)

Niech $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ będzie kwerendą, c żadaną dokładnością odpowiedzi. Odpowiedzią na kwerendę q z dokładnością c nazywać będziemy

$$f(q, c) = \{(d, v) : (d, v) \in \bigcap_{i=1}^k (LR(t_i) \mid_{cw_i})\}. \quad (14)$$

■

Dla każdej pary (d, v) należącej do odpowiedzi wartość v jest stopniem podobieństwa kwerendy q do opisu dokumentu $t(d)$.

Przykład 7

Niech zbiór deskryptorów i opisy dokumentów będą takie same jak w przykładzie 4. Uporządkowane listy inwersyjne uwzględniające relację uogólnienia R mają postać:

$$LR(t_1) = \{(d_1, 1.0), (d_2, 1.0), (d_3, 1.0), (d_4, 1.0), (d_5, 1.0)\}$$

$$LR(t_2) = \{(d_1, 0.4), (d_4, 0.1)\}$$

$$LR(t_3) = \{(d_2, 0.7), (d_3, 0.4), (d_4, 0.4), (d_1, 0.3)\}$$

$$LR(t_4) = \{(d_5, 1.0), (d_3, 0.6), (d_4, 0.5), (d_1, 0.3), (d_2, 0.3)\}$$

$$LR(t_5) = \{(d_5, 0.4), (d_1, 0.3)\}$$

$$LR(t_6) = \{(d_5, 0.6)\}.$$

Niech kwerenda będzie postaci $q = \{(t_3, 0.5), (t_4, 0.5)\}$ i wymagana dokładność odpowiedzi wynosić $c = 0.8$. Odpowiedź na kwerendę q z dokładnością c ma postać:

$$\begin{aligned} f(q, 0.8) &= LR(t_3) \Big|_{0.8*0.5} \cap LR(t_4) \Big|_{0.8*0.5} = \\ &= \{(d_2, 0.7), (d_3, 0.4), (d_4, 0.4), (d_1, 0.3)\} \Big|_{0.8*0.5} \cap \\ &\quad \cap \{(d_5, 1.0), (d_3, 0.6), (d_4, 0.5), (d_1, 0.3), (d_2, 0.3)\} \Big|_{0.8*0.5} \\ &= \{(d_3, \min(0.5, 0.4) + \min(0.5, 0.6)), (d_4, \min(0.5, 0.4) + \min(0.5, 0.5))\} \\ &= \{(d_3, 0.9), (d_4, 0.9)\}. \end{aligned}$$

Do odpowiedzi nie należą dokumenty d_1 i d_2 , ponieważ wagi tych dokumentów na listach inwersyjnych są mniejsze od wartości progowej 0.4. Ponadto, sprawdzając opisy dokumentów d_1 i d_2 , można zauważyć, że nie jest również spełniony warunek (7) w wyrażeniu (10) dla opisów tych dokumentów.

Twierdzenie 10

Niech $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ będzie kwerendą.

Jeśli $(d, v) \in \bigcap_{i=1}^k LR(t_i) \Big|_{cw_i}$, to $v \geq c$.

Dowód

Niech $(d, v_i) \in LR(t_i) \Big|_{cw_i}$, stąd $cw_i \leq v_i$.

Na podstawie nierówności $cw_i \leq v_i$ i warunku $\sum_{i=1}^k w_i = 1$ otrzymujemy

$$c = cw_1 + cw_2 + \dots + cw_k \leq \min(v_1, w_1) + \min(v_2, w_2) + \dots + \min(v_k, w_k) = \mu(q, t(d)) = v.$$

Podczas obliczania przekrojów list inwersyjnych nie korzystamy z pełnych list inwersyjnych, wybieramy z nich tylko te dokumenty, które spełniają warunek $\mu(q, t(d)) \geq c$. Stąd można przypuszczać, że przy wymaganych dużych dokładnościach odpowiedzi, czas wyszukiwania może być krótszy niż w systemach bez wag deskryptorów.

W wielu systemach wyszukiwania informacji strategię przypisywania wag deskryptorom zależą od wielu czynników [11], [87] i bardzo często opis dokumentu nie spełnia warunku $\sum w_i \leq 1$, wtedy należy go znormalizować, np. każdą wagę deskryptora podzielić przez sumę wszystkich wag. Wadą takiego rozwiązania jest możliwość powstania jednakowych opisów dla różnych dokumentów. Jeśli wagi deskryptorów są dowolne, wszystkie opisane własności modelu systemu zostaną zachowane, gdy używać będziemy znormalizowanej miary podobieństwa

$$\mu^*(t_1, t_2) = \mu(t_1, t_2) / \sum_{i=1}^n w_{1i}$$

oraz gdy definicja 5 przyjmie następującą postać:

Definicja 5'

Niech $(d_i, v) \in L(t_1) \big|_{cw_1}$ i $(d_k, v_k) \in L(t_2) \big|_{cw_2}$.

$$(d, v) \in L(t_1) \big|_{cw_1} \cap L(t_2) \big|_{cw_2} \Leftrightarrow$$

$$\Leftrightarrow \forall_{\substack{d_i = d_k \\ v}} (d, v) = (d_i, (\min(v_i, w_1) + \min(v_k, w_2)) / (w_1 + w_2)) \wedge v_i \geq cw_1 \wedge v_k \geq cw_2.$$

3. WŁASNOŚCI ROZPROSZONEGO SYSTEMU WYSZUKIWANIA INFORMACJI

Rozważmy teraz problem połączenia n lokalnych systemów wyszukiwania informacji $S_1, S_2, \dots, S_n$ w jeden rozproszony system S . Każdy lokalny system $S_j = (T_j, D_j, Q_j, f_j)$, ($1 \leq j \leq n$) ma wszystkie własności podane w rozdziale 2. Na bazie ciągu systemów $\{S_j\}$ ($1 \leq j \leq n$) możemy zbudować rozproszony system wyszukiwania informacji oparty na tezaursie z wagami [50].

Definicja 7

Rozproszonym systemem wyszukiwania informacji opartym na tezaursie z wagami nazywamy system $S = (T, D, Q, f)$, w którym

- (i) $T = \bigcup_{j=1}^n T_j$, $\phi_j = \phi \cap (T_j \times T_j)$, $R_j = R \cap (T_{0j} \times T_{0j})$,
- (ii) $D = \bigcup_{j=1}^n D_j$,
- (iii) $Q = \bigcup_{j=1}^n Q_j$, $\gamma_{jc} = \gamma_c \cap (Q_j \times Q_j)$,
- (iv) $\forall f(q, c) = \{(d, v) : d \in D \wedge q \gamma_c t(d) \wedge v = \mu(q, t(d))\}$,
 $q \in Q$

Warunki te mówią, że w rozproszonym systemie zbiory dokumentów i zbiory deskryptorów są sumami dokumentów i deskryptorów z systemów składowych. Aby złączenie systemów było dobrze określone, musi być spełniony warunek:

Jeśli $d \in D_j \cap D_k \wedge t \in T_j \cap T_k \wedge (t, w) \in t_j(d)$, to $(t, w) \in t_k(d)$ (15)

Warunek (15) mówi, że jeśli w systemach składowych S_j, S_k występują jednakowe dokumenty i jednakowe deskryptory, to opisy dokumentów w tych podsystemach muszą zawierać te same deskryptory i z tymi samymi wagami. Jest to naturalny warunek i nie wymaga on dalszego wyjaśnienia. Zbiory T_j ($1 \leq j \leq n$) nie muszą być rozłączne. Deskryptory ze zbiorów T_j i T_k

mogą opisywać ten sam dokument należący do zbiorów D_j i $D_{j''}$. Na przykład, jeśli w rozproszonym systemie S opis dokumentu $d \in D$ jest postaci: $t(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_i, w_i), (t_{i+1}, w_{i+1}), \dots, (t_k, w_k), (t_{k+1}, w_{k+1}), \dots, (t_m, w_m)\}$, gdzie $t_1, t_2, \dots, t_k \in T_j$ i $t_{i+1}, \dots, t_m \in T_{j''}$, to dokument d należący jednocześnie do zbiorów D_j i $D_{j''}$ będzie miał w podsystemie S_j opis postaci $t_j(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$, natomiast w podsystemie $S_{j''}$ opis $t_{j''}(d) = \{(t_{i+1}, w_{i+1}), \dots, (t_m, w_m)\}$. Relacja zawierania się $\succ_c$ jest zdefiniowana dla opisów dokumentów tak samo jak dla kwerend.

Definicja 8

Globalna kwerenda $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\} \in Q$ należy do zbioru Q_j wtedy i tylko wtedy, gdy $\forall_{1 \leq i \leq k} X_i = \{t: t \in T_0 \wedge t_1 R t\} \subset T_{0j}$.

■

Można pokazać istnienie pewnych algebraicznych związków pomiędzy ciągiem systemów $\{S_j\}$ a rozproszonym systemem wyszukiwania informacji zbudowanym na ich podstawie. Zdefiniujemy na początek pojęcie podsystemu.

Definicja 9

Niech $S_1 = (T_1, D_1, Q_1, f_1)$ i $S_2 = (T_2, D_2, Q_2, f_2)$ będą systemami wyszukiwania informacji. System S_1 jest podsystemem systemu S_2 , jeśli:

- (i) $T_1 \subset T_2 \wedge R_1 = R_2 \cap (T_{01} \times T_{01}) \wedge \phi_1 = \phi_2 \cap (T_1 \times T_1)$,
- (ii) $D_1 \subset D_2$,
- (iii) $Q_1 \subset Q_2 \wedge \succ_{1c} = \succ_{2c} \cap (Q_1 \times Q_1)$,
- (iv) $f_1(q, c) = f_2(q, c) \cap D_1$ dla $q \in Q_1$.

Twierdzenie 11

System wyszukiwania informacji $S_j = (T_j, D_j, Q_j, f_j)$ dla każdego j ($1 \leq j \leq n$), jest podsystemem rozproszonego systemu wyszukiwania informacji $S = (T, D, Q, f)$.

Dowód

Aby udowodnić twierdzenie, musimy sprawdzić wszystkie warunki definicji 9

(i) ponieważ $T = \bigcup_{j=1}^n T_j$, stąd dla każdego j ($1 \leq j \leq n$) zachodzi $T_j \subset T$, pozostałe elementy warunku wynikają z definicji 7(i),

(ii) z równości $D = \bigcup_{j=1}^n D_j$ wynika, że dla każdego $1 \leq j \leq n$, $D_j \subset D$,

(iii) $Q \supset \bigcup_{j=1}^n Q_j$, stąd $Q_j \subset Q$ dla każdego j , druga część warunku wynika z definicji 7(iii),

(iv) należy udowodnić, że

$$\forall_{1 \leq j \leq n} \quad \forall_{q \in Q_j} \quad f(q, c) \cap D_j = f_j(q, c).$$

Ustalmy dowolne j_0 ($1 \leq j_0 \leq n$). Korzystając z definicji 9(ii)-(iv) otrzymujemy, że

$$\begin{aligned} \forall_{q \in Q_{j_0}} \quad f(q, c) \cap D_{j_0} &= \{ (d, v) : d \in D \wedge q \succ_c t(d) \wedge v = \mu(q, t(d)) \} \cap D_{j_0} = \\ &= \{ (d, v) : d \in D_{j_0} \wedge q \succ_c t(d) \wedge v = \mu(q, t(d)) \} = \\ &= \{ (d, v) : d \in D_{j_0} \wedge q \succ_{j_0} t_{j_0}(d) \wedge v = \mu(q, t_{j_0}(d)) \}. \end{aligned}$$

Ponieważ j_0 było dowolne, więc $\forall_{1 \leq j \leq n} \quad \forall_{q \in Q_j} \quad f(q, c) \cap D_j = f_j(q, c).$

■

Na podstawie na warunku (15) łatwo można zauważyć, że globalny opis dokumentu $t(d)$ w systemie S jest sumą mnogościową opisów tego dokumentu w lokalnych podsystemach, tj.

$$t(d) = \bigcup_{j=1}^n t_j(d). \quad (16)$$

Ponieważ w rozproszonym systemie zbiór dokumentów jest sumą lokalnych zbiorów dokumentów, w dalszych rozważaniach spróbujemy wyrazić odpowiedź na dowolną kwerendę jako wynik pewnych operacji na odpowiedziach uzyskanych z lokalnych systemów wyszukiwania informacji.

Niech f_j^* będzie odpowiedzią z wymaganą dokładnością $c=0$ na kwerendę $q = \{ (t, w) \}$ zawierającą pojedynczy deskryptor t z wagą w

$$f_j^*(t) = \{ (d, v) : d \in D_j \wedge (t, w) \in t_j(d) \wedge v = w \} \quad (17)$$

Dokumenty zawierające w swoim opisie deskryptor t z wagą większą lub równą wartości progowej c_w , będą należały w podsystemie S_j do częściowej odpowiedzi $f_j^*(t)$. Waga dokumentu równa jest wadze deskryptora t z jaką wystąpił on w opisie tego dokumentu. Łatwo zauważyć, że jeśli $t \notin T_j$, to $f_j^*(t) = \emptyset$.

Przed podaniem definicji globalnej odpowiedzi na dowolną kwerendę zdefiniujemy operacje sumy " $\bar{\cup}$ " i przecięcia " $\cap$ " częściowych odpowiedzi na kwerendy zawierające pojedyncze deskryptory.

Definicja 10

Niech $q_1 = \{(t_1, w_1)\}$, $q_2 = \{(t_2, w_2)\}$ będą kwerendami i niech $(d', v') \in f_j^*(t_1)$, $(d'', v'') \in f_j^*(t_2)$.

$$(i) \quad (d, v) \in f_j^*(t_1) \bar{\cup} f_j^*(t_2) \iff \exists_{d' \neq d''} \left[(d, v) = (d', v') \vee (d, v) = (d'', v'') \right] \\ \vee \exists_{d'=d''} (d, v) = (d', v' + v''),$$

$$(ii) \quad (d, v) \in f_j^*(t_1) \cap f_j^*(t_2) \iff \forall_{d'=d''} (d', \min(w_1, v') + \min(w_2, v'')).$$

Niech

$$q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\} \in Q \quad (18)$$

będzie globalną kwerendą i niech $q_i = \{(t_i, w_i)\}$, $(1 \leq i \leq k)$ będzie częściową kwerendą, powstałą w procesie dekompozycji globalnej kwerendy, zawierającą pojedynczy deskryptor. Oznaczmy przez X_i zbiór wszystkich deskryptorów bardziej szczegółowych od deskryptora t_i , tj.

$$\forall_{t_i \in T_0} X_i = \{t: t \in T_0 \wedge (t_i = t \vee t_i R t)\}.$$

Definicja 11

W rozproszonym systemie S odpowiedź na globalną kwerendę (18) z dokładnością c ma postać

$$f(q, c) = \bigcap_{i=1}^k \left\{ (d, v_i) \in \left[\bigcup_{t \in X_i} \bigcup_{j=1}^n f_j^*(t) \right] : v_i \geq w_i c \right\} \quad (19)$$

Po odpowiednich przekształceniach możemy pokazać, że definicja 7(iv) odpowiedzi w rozproszonym systemie S jest zgodna z postacią odpowiedzi w definicji 11.

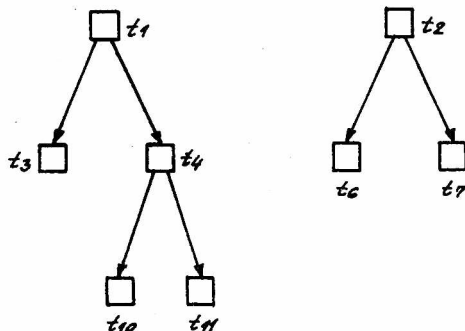
Niech $W_i = \{w' : (t', w') \in (d) \wedge t' \in X_i\}$.

$$f(q, c) = \{(d, v) : d \in D \wedge q \succ_c t(d) \wedge v = \mu(q, t(d))\} =$$

$$\begin{aligned} &= \left\{ (d, v) : d \in D \wedge \bigvee_{1 \leq i \leq k} \exists (t_i, w) \in t(d) \quad (t_i = t \vee t_i R t) \wedge w_i c \leq v_i \wedge \right. \\ &\quad \left. \wedge v_i = \sum_{w' \in W_i} w' \wedge v = \sum_{i=1}^k \min(w_i, v_i) \right\} = \\ &= \bigcap_{i=1}^k \left\{ (d, v_i) : d \in D \wedge \{(t_i, w_i)\} \succ_c t(d) \wedge v_i = \sum_{w' \in W_i} w' \right\} = \\ &= \bigcap_{i=1}^k \left\{ (d, v_i) \in \bigcup_{\substack{t \in X_i \\ t \in X_i}} \{(d, w) : d \in D \wedge (t, w) \in t(d)\} : v_i \geq w_i c \right\} = \\ &= \bigcap_{i=1}^k \left\{ (d, v_i) \in \left(\bigcup_{t \in X_i} \bigcap_{j=1}^n \{(d, w) : d \in D_j \wedge (t, w) \in t_j(d)\} \right) : v_i \geq w_i c \right\} = \\ &= \bigcap_{i=1}^k \left\{ (d, v_i) \in \left(\bigcup_{t \in X_i} \bigcap_{j=1}^n f_j^*(t) \right) : v_i \geq w_i c \right\}. \end{aligned}$$

Przykład 8

Niech będą dane dwa lokalne systemy $S_1 = (T_1, D_1, Q_1, f_1)$ i $S_2 = (T_2, D_2, Q_2, f_2)$. Niech $T_1 = \{t_1, t_2, t_3, t_4, t_6, t_7, t_{10}, t_{11}\}$ będzie tezauresem (rys.3) w systemie S_1 oraz $t_1 R_1 t_3$, $t_1 R_1 t_4$, $t_4 R_1 t_{11}$, $t_4 R_1 t_{10}$, $t_2 R_1 t_6$, $t_2 R_1 t_7$.



Rys.3. Graf opisujący relację uogólnienia w systemie S_1
Fig.3. Graph describing generalization relation in system S_1

Załóżmy, że opisy dokumentów ze zbioru $D_1 = \{d_1, d_2, d_3, d_4, d_5, d_7\}$ mają postać:

$$t_1(d_1) = \{(t_6, 0.4), (t_7, 0.4)\}$$

$$t_1(d_2) = \{(t_4, 0.5)\}$$

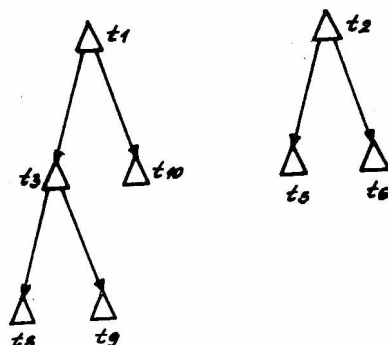
$$t_1(d_3) = \{(t_2, 0.2), (t_4, 0.8)\}$$

$$t_1(d_4) = \{(t_7, 0.5), (t_{10}, 0.2)\}$$

$$t_1(d_5) = \{(t_4, 0.6), (t_7, 0.4)\}$$

$$t_1(d_7) = \{(t_{10}, 0.2)\}.$$

Odpowiedź w lokalnym systemie S_1 na kwerendę $q = \{(t_1, 0.6), (t_2, 0.4)\}$ z wymaganą dokładnością $c = 0.5$ ma postać $f_1(q, 0.5) = \{(d_3, 0.8), (d_5, 1.0)\}$. Odpowiednio w systemie S_2 niech $T_2 = \{t_1, t_2, t_3, t_5, t_6, t_8, t_9, t_{10}\}$ będzie tezauresem (rys. 4), w którym $t_1 R_2 t_3$, $t_1 R_2 t_{10}$, $t_3 R_2 t_8$, $t_3 R_2 t_9$, $t_2 R_2 t_5$, $t_2 R_2 t_6$.



Rys. 4. Graf opisujący relację uogólnienia w systemie S_2

Fig. 4. Graph describing generalization relation in system S_2

Założmy ponadto, że dokumenty ze zbioru $D_2 = \{d_1, d_2, d_4, d_6, d_7\}$ mają następujące opisy:

$$t_2(d_1) = \{(t_5, 0.2), (t_6, 0.4)\}$$

$$t_2(d_2) = \{(t_5, 0.5)\}$$

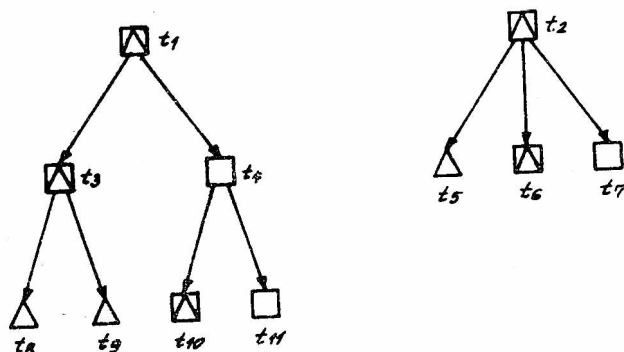
$$t_2(d_4) = \{(t_5, 0.3), (t_{10}, 0.2)\}$$

$$t_2(d_6) = \{(t_5, 0.4), (t_9, 0.5), (t_8, 0.1)\}$$

$$t_2(d_7) = \{(t_8, 0.4), (t_{10}, 0.2)\}.$$

W systemie S_2 odpowiedź na tę samą kwerendę $q = \{(t_1, 0.6), (t_2, 0.4)\}$ z dokładnością $c = 0.5$ ma postać $f_2(q, 0.5) = \{(d_6, 1.0)\}$.

Na bazie lokalnych systemów S_1 i S_2 możemy skonstruować rozproszony system $S=(T,D,Q,f)$, w którym $T=T_1 \cup T_2$ jest tezauresem z relacją uogólnienia R spełniającą warunki: $R_1=R \cap (T_{01} \times T_{01})$ i $R_2=R \cap (T_{02} \times T_{02})$, $D=D_1 \cup D_2$ zbiorem dokumentów, Q zbiorem kwerend i $f:Q \times [0,1] \rightarrow 2^{D \times [0,1]}$. Globalny tezaurus (rys. 5) jest zbiorem deskryptorów $T=\{t_1, t_2, \dots, t_{11}\}$, w którym zachodzą relacje $t_1 Rt_3$, $t_1 Rt_4$, $t_3 Rt_8$, $t_3 Rt_9$, $t_4 Rt_{10}$, $t_4 Rt_{11}$, $t_2 Rt_5$, $t_2 Rt_6$, $t_2 Rt_7$.



Rys. 5. Postać globalnego tezausa T
Fig. 5. Structure of the global thesaurus T

Na rysunku 5 deskryptory należące do zbioru T_1 zaznaczono kwadracikami, a deskryptory należące do zbioru T_2 trójkącikami.

Zgodnie z warunkiem (16), globalne opisy dokumentów ze zbioru $D=\{d_1, d_2, \dots, d_7\}$ mają następującą postać:

$$t(d_1)=\{(t_5, 0.2), (t_6, 0.4), (t_7, 0.4)\}$$

$$t(d_2)=\{(t_4, 0.5), (t_5, 0.5)\}$$

$$t(d_3)=\{(t_2, 0.2), (t_4, 0.8)\}$$

$$t(d_4)=\{(t_7, 0.5), (t_5, 0.3), (t_{10}, 0.2)\}$$

$$t(d_5)=\{(t_4, 0.6), (t_7, 0.4)\}$$

$$t(d_6)=\{(t_5, 0.4), (t_9, 0.5), (t_8, 0.1)\}$$

$$t(d_7)=\{(t_5, 0.8), (t_{10}, 0.2)\}.$$

Zbiory deskryptorów bardziej szczegółowych od deskryptorów t_1 i t_2 mają postać:

$$X_1=\{t \in T: t_1=t \vee t_1 Rt\}=\{t_1, t_3, t_4, t_8, t_9, t_{10}, t_{11}\},$$

$$X_2=\{t \in T: t_2=t \vee t_2 Rt\}=\{t_2, t_5, t_6, t_7\}.$$

Zgodnie z definicją 11, w rozproszonym systemie odpowiedź na kwerendę $q = \{(t_1, 0.6), (t_2, 0.4)\}$ z dokładnością $c = 0.5$ wynosi

$$\begin{aligned} f(q, c) &= \left\{ (d, v_1) \in \bigcup_{t \in X_1} (f_1^*(t) \cup f_2^*(t)) : v \geq 0.6 * 0.5 \right\} \cap \\ &\quad \cap \left\{ (d, v_2) \in \bigcup_{t \in X_2} (f_1^*(t)) : v_2 \geq 0.4 * 0.5 \right\} = \\ &= \left\{ (d_2, 0.5), (d_3, 0.8), (d_5, 0.6), (d_6, 0.6) \right\} \cap \left\{ (d_1, 1.0), (d_2, 0.5), \right. \\ &\quad \left. (d_3, 0.2), (d_4, 0.8), (d_5, 0.4), (d_6, 0.4), (d_7, 0.4) \right\} = \\ &= \left\{ (d_2, 0.5), (d_3, 0.8), (d_5, 1.0), (d_6, 1.0) \right\}. \end{aligned}$$

Z definicji 11 wynika bezpośrednio poniższe twierdzenie.

Twierdzenie 12

Jeśli $c_1 \geq c_2$, to $f(q, c_1) \subset f(q, c_2)$.

Podstawową własność odpowiedzi, a mianowicie zależność zachodzącą pomiędzy dokładnością c a stopniem zawierania się $v = \mu(q, t)$ kwerendy q w opisie dokumentu należącego do odpowiedzi f , można wyrazić w postaci następującego twierdzenia.

Twierdzenie 13

Niech $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ będzie kwerendą.

Jeśli $(d, v) \in f(q, c)$, to $v \geq c$.

Dowód

Dla każdego i ($1 \leq i \leq k$) prawdziwe są równości:

$$X_i = \{t \in T_0 : t_i = t \vee t_i R t\} = \bigcup_{j=1}^n X_{ij} = \bigcup_{j=1}^n \{t \in T_{0j} : t_i = t \vee t_i R_j t\}.$$

Niech $W_i = \{w' : (t', w') \in \bigcup_{j=1}^n t_j(d) \wedge t' \in X_i\}$ i $v_i = \sum_{w' \in W_i} w'$.

Z nierówności $\forall_{(t_i, w_i) \in q} w_i c \leq v_i$ wynikającej z wyrażenia (19) oraz

nierówności $\forall_{0 \leq c \leq 1} w_i c \leq w_i$ otrzymujemy, że $w_i c \leq \min(w_i, v_i)$.

Ponieważ $\sum_{i=1}^k w_i = 1$, stąd $c = \sum_{i=1}^k w_i c \leq \sum_{i=1}^k \min(w_i, v_i) = \mu(q, t(d)) = v$.

■

Twierdzenie odwrotne nie jest prawdziwe, tzn. jeśli $v \geq c$, to nie zawsze $(d, v) \in (q, c)$.

W pewnych szczególnych przypadkach, przy dodatkowych założeniach, operacje na odpowiedziach z lokalnych systemów mogą być uproszczone [56]:

(i) Wszystkie podsystemy składowe systemu rozproszonego mają jednakowe tezaury, lecz różne zbiory dokumentów. Odpowiedź można przedstawić jako sumę lokalnych odpowiedzi. Jeśli dla każdego j ($1 \leq j \leq n$) $T_j = T$, to w rozproszonym systemie S globalna odpowiedź ma postać $f(q, c) = \bigcup_{j=1}^n f_j(q, c)$. Ponieważ wszystkie podsystemy mają jednakowe tezaury, postępowanie takie jest dopuszczalne.

(ii) Wszystkie podsystemy systemu rozproszonego mają rozłączne zbiory dokumentów. Globalną odpowiedź (19) można przedstawić jako sumę lokalnych odpowiedzi

$$f(q, c) = \bigcup_{j=1}^n \left\{ \bigcap_{i=1}^k \left\{ (d, v) \in \bigcup_{t \in X_i} f_j^*(t) : v \geq w_i c \right\} \right\} = \bigcup_{j=1}^n f_j(q, c).$$

Projektując algorytmy sprawnej realizacji ciągów operacji na lokalnych odpowiedziach, można rozważać rozmaite struktury danych reprezentujące lokalne odpowiedzi. W pracy [1] zwrócono uwagę na czasową złożoność ciągu $\mathcal{Q}$ składającego się z podstawowych operacji na zbiorach, tj. na czas wymagany do wykonania wszystkich instrukcji w $\mathcal{Q}$, mierzony jako funkcja długości ciągu $\mathcal{Q}$ i rozmiaru bazy danych. W pracy tej rozważa się zarówno średnią, jak i pesymistyczną złożoność czasową oraz rozróżnia się złożoność on-line i off-line.

4. ORGANIZACJA LIST INWERSYJNYCH W ROZPROSZONYM SYSTEMIE WYSZUKIWANIA INFORMACJI

Dla każdego deskryptora $t \in T_j$ lista inwersyjna $L_j(t)$ w podsystemie S_j jest zbiorem par (dokument, waga). Dokument d , zawierający parę (t, w) w swoim opisie $t_j(d)$, tworzy wraz z wagą w element listy $L_j(t)$

$$\forall_{t \in T_j} L_j(t) = \{(d, v) : d \in D_j \wedge (t, w) \in t_j(d) \wedge v = w\}.$$

W każdym podsystemie S_j częściowe odpowiedzi na kwerendy $q = \{(t_1, w_1)\}$ zawierające pojedyncze deskryptory, zgodnie z wyrażeniami (12) i (16), mają postać:

$$f_j^*(t_1) = L_j(t_1), \quad (\text{jeśli } t_1 \in T_j, \text{ to } L_j(t_1) = \emptyset).$$

W rozproszonym systemie odpowiedź na globalną kwerendę można przedstawić jako wynik pewnych operacji dokonywanych na odpowiednich listach inwersyjnych z lokalnych podsystemów wyszukiwania informacji. Zgodnie z definicją 11, odpowiedź na dowolną kwerendę $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\} \in Q$ z dokładnością c możemy wyrazić jako

$$f(q, c) = \bigcap_{i=1}^k \left[\bigcup_{t \in X_i} \bigcup_{j=1}^n L_j(t) \right] \Big|_{cw_i} \quad (20)$$

Przykład 9

Niech S_1, S_2 będą systemami z przykładu 8. Dla poszczególnych deskryptorów t listy inwersyjne $L_j(t)$ mają postać:

(i) w podsystemie S_1

$$L_1(t_1) = \emptyset$$

$$L_1(t_2) = \{(d_3, 0.2)\}$$

$$L_1(t_3) = \emptyset$$

$$L_1(t_4) = \{(d_2, 0.5), (d_3, 0.8), (d_5, 0.6)\}$$

$$L_1(t_6) = \{(d_1, 0.4)\}$$

$$L_1(t_7) = \{(d_1, 0.4), (d_4, 0.5), (d_5, 0.4)\}$$

$$L_1(t_{10}) = \{(d_4, 0.2), (d_7, 0.2)\}$$

$$L_1(t_{11}) = \emptyset.$$

(ii) w podsystemie S_2

$$L_2(t_1) = \emptyset$$

$$L_2(t_2) = \emptyset$$

$$L_2(t_3) = \emptyset$$

$$L_2(t_5) = \{(d_1, 0.2), (d_2, 0.5), (d_4, 0.3), (d_6, 0.4), (d_7, 0.4)\}$$

$$L_2(t_6) = \{(d_1, 0.4)\}$$

$$L_2(t_8) = \{(d_6, 0.1)\}$$

$$L_2(t_9) = \{(d_6, 0.5)\}$$

$$L_2(t_{10}) = \{(d_4, 0.2), (d_7, 0.2)\}.$$

Niech $q = \{(t_1, 0.6), (t_2, 0.4)\}$ będzie globalną kwerendą i $c = 0.5$ będzie wymaganą dokładnością odpowiedzi. Zbiory deskryptorów ściślejszych od deskryptorów t_1 i t_2 mają postać:

$$X_1 = \{t: t \in T \wedge (t_1 = t \vee t_1 R t)\} = \{t_1, t_3, t_4, t_8, t_9, t_{10}, t_{11}\}$$

$$X_2 = \{t: t \in T \wedge (t_2 = t \vee t_2 R t)\} = \{t_2, t_5, t_6, t_7\}.$$

Zgodnie z definicją 11, w rozproszonym systemie odpowiedź ma postać

$$\begin{aligned} f(q, c) &= \left[\bigcup_{t \in X_1} (L_1(t) \cup L_2(t)) \right] \Big|_{0.6 * 0.5} \cap \left[\bigcup_{t \in X_2} (L_1(t) \cup L_2(t)) \right] \Big|_{0.4 * 0.5} = \\ &= \left[\{(d_2, 0.5), (d_3, 0.8), (d_5, 0.6)\} \cup \{(d_6, 0.1)\} \cup \{(d_6, 0.5)\} \right] \Big|_{0.6 * 0.5} \\ &\quad \cap \left[\{(d_3, 0.2)\} \cup \{(d_1, 0.2), (d_2, 0.5), (d_4, 0.3), (d_6, 0.4), (d_7, 0.4)\} \right. \\ &\quad \left. \cup \{(d_1, 0.4)\} \cup \{(d_1, 0.4), (d_4, 0.5), (d_5, 0.4)\} \right] \Big|_{0.4 * 0.5} = \\ &= \{(d_2, 0.5), (d_3, 0.8), (d_5, 0.6), (d_6, 0.6)\} \Big|_{0.6 * 0.5} \cap \{(d_3, 0.3), (d_1, 1.0), \\ &\quad (d_2, 0.5), (d_4, 0.8), (d_5, 0.4), (d_6, 0.4), (d_7, 0.4)\} \Big|_{0.4 * 0.5} = \\ &= \{(d_2, \min(0.5, 0.6) + \min(0.5, 0.4)), (d_3, \min(0.8, 0.6) + \min(0.2, 0.4)), \end{aligned}$$

$$(d_5, \min(0.6, 0.6) + \min(0.4, 0.4)), (d_6, \min(0.6, 0.6) + \min(0.4, 0.4)) \} = \\ = \{ (d_2, 0.9), (d_3, 0.8), (d_5, 1.0), (d_6, 1.0) \}.$$

■

Twierdzenie 14

Niech $q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ będzie kwerendą.

$$\text{Jeśli } (d, v) \in (q, c) = \bigcap_{i=1}^k \left(\bigcup_{t \in X_i} \bigcup_{j=1}^n L_j(t) \right) \Big|_{cw_i}, \text{ to } v \geq c.$$

Dowód

$$\text{Jeśli } (d, v) \in \bigcap_{i=1}^k \left(\bigcup_{t \in X_i} \bigcup_{j=1}^n L_j(t) \right) \Big|_{cw_i}, \text{ to } (d, v) \in \left(\bigcup_{t \in X_i} \bigcup_{j=1}^n L_j(t) \right) \Big|_{cw_i}.$$

Z warunku $\sum_{i=1}^k w_i = 1$ i definicji list inwersyjnych otrzymujemy

$$v = \sum_{i=1}^k \min(w_i, v_i) \geq \sum_{i=1}^k \min(w_i, v_i) \geq \sum_{i=1}^k cw_i = c.$$

■

Metoda list inwersyjnych zachowuje własność inkluzyjności systemu opartego na tezaurusach.

Lemma

Niech $q_1 = \{(t_1, w_1)\}$, $q_2 = \{(t_2, w_2)\}$ będą kwerendami zawierającymi pojedyncze deskryptory.

Jeśli $q_1 \succ_c q_2 \wedge c_1 \leq c_2$, to

$$f(q_2, c_2) = \left(\bigcup_{t \in X_2} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_2 w_2} \subset \left(\bigcup_{t \in X_1} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_1} = f(q_1, c_1).$$

Dowód

Jeśli $q_1 \succ_c q_2$, to $(t_1 = t_2 \vee t_1 R t_2) \wedge w_1 c \leq w_2$, stąd wynika, że

$$X_1 = \{t \in T: t_1 = t \vee t_1 R t\} \supset \{t \in T: t_2 = t \vee t_2 R t\} = X_2.$$

Niech $X = X_1 \setminus X_2$, stąd

$$\left(\bigcup_{t \in X_1} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_1} = \left(\bigcup_{t \in X_2} \bigcup_{j=1}^n L_j(t) \cup \bigcup_{t \in X} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_1}.$$

Niech $(d, v) \in \left(\bigcup_{t \in X_2} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_1} = f(q_2, c_2)$.

Korzystając z nierówności $v \geq c_2$ i $v \geq c_2 w_2 \geq c_2 c w_1 \geq c_1 w_1$ otrzymujemy

$$(d, v) \in f(q_1, c_1) = \left(\bigcup_{t \in X_1} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_1}.$$

Twierdzenie 15

Niech

$$q_1 = \{ (t_{11}, w_{11}), (t_{12}, w_{12}), \dots, (t_{1k_1}, w_{1k_1}) \}$$

$$q_2 = \{ (t_{21}, w_{21}), (t_{22}, w_{22}), \dots, (t_{2k_2}, w_{2k_2}) \}$$

będą kwerendami.

Jeśli $q_1 \succ_c q_2 \wedge c_1 \leq c_2 c$, to

$$f(q_2, c_2) = \bigcap_{i_2=1}^{k_2} \left(\bigcup_{t \in X_2} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_2 w_{i_2}} < \bigcap_{i_1=1}^{k_1} \left(\bigcup_{t \in X_1} \bigcup_{j=1}^n L_j(t) \right) \Big|_{c_1 w_{i_1}} = f(q_1, c_1)$$

Dowód

Niech $q_{1i_1} = \{ (t_{1i_1}, w_{1i_1}) \}$, $q_{2i_2} = \{ (t_{2i_2}, w_{2i_2}) \}$, ($1 \leq i_1 \leq k_1$, $1 \leq i_2 \leq k_2$) będą częściowymi kwerendami.

Z definicji relacji $\succ_c$ wynika, że jeśli $q_1 \succ_c q_2$, to

$$\forall_{1 \leq i_1 \leq k_1} \exists_{1 \leq i_2 \leq k_2} (t_{1i_1} = t_{2i_2} \vee t_{1i_1} R t_{2i_2}) \text{ i } w_{1i_1} c \leq w_{2i_2}.$$

Korzystając z Lematu możemy napisać

$$\forall_{1 \leq i_1 \leq k_1} \exists_{1 \leq i_2 \leq k_2} f(q_{1i_1}, c_1) \supset f(q_{2i_2}, c_2).$$

Oznaczamy przez $A = \{ i_2 : 1 \leq i_2 \leq k_2 \wedge \exists_{1 \leq i_1 \leq k_1} (t_{1i_1} = t_{2i_2} \vee t_{1i_1} R t_{2i_2}) \}$.

Jeśli $(d, v) = \bigcap_{i_2=1}^{k_2} f(q_{2i_2}, c_{2i_2})$, to

$$\begin{aligned}
 v &= \sum_{i_2=1}^{k_2} \min(w_{2i_2}, v_{i_2}) \geq \sum_{i_2 \in A} \min(w_{2i_2}, v_{i_2}) \geq \sum_{i_2 \in A} \min(w_{2i_2}, c_{2i_2}) \geq \sum_{i_2 \in A} c_{2i_2} w_{2i_2} \geq \\
 &\geq \sum_{i_1=1}^{k_1} c_{2i_1} w_{1i_1} \geq \sum_{i_1=1}^{k_1} c_{1i_1} w_{1i_1},
 \end{aligned}$$

stąd wynika, że $(d, v) \in \bigcap_{i_1=1}^{k_1} f(q_{1i_1}, c_{1i_1})$.

■

Z definicji odpowiedzi wynika, że w rozproszonym systemie nie trzeba tworzyć nowych globalnych list inwersyjnych, można natomiast wykorzystać istniejące lokalne listy inwersyjne w podsystemach. W rozproszonym systemie opartym na tezaurusach z wagami proces wyszukiwania przebiega tak samo jak w metodzie list inwersyjnych, różnica polega na wykonaniu dodatkowej operacji sumowania odpowiednich list inwersyjnych z lokalnych podsystemów.

5. HETEROGENICZNE SYSTEMY WYSZUKIWANIA INFORMACJI

Kładąc warunki definicji tezauryśa podanej w rozdziale 2, możemy otrzymać nowe systemy wyszukiwania informacji. Zastępując relację uogólnienia relacją równości określoną na zbiorze deskryptorów, otrzymujemy wagowy model rozproszonego systemu wyszukiwania informacji [48]. Po usunięciu wag z opisów dokumentów otrzymujemy rozproszony system wyszukiwania informacji oparty na tezaursach [56]. Natomiast gdy w tezaursie określona jest tylko relacja równości deskryptorów i w opisach dokumentów nie występują wagi deskryptorów, wtedy otrzymujemy prosty rozproszony system wyszukiwania informacji, oparty na zbiorach deskryptorów [49]. Na bazie tych modeli systemów możemy zbudować nowe rozproszone systemy wyszukiwania informacji.

5.1. Rozproszony system wyszukiwania informacji oparty na tezaursach

W każdym lokalnym systemie $S_j = (T_j, D_j, Q_j, f_j)$, opisanym w pracach [29], [96], tezaurs T_j jest zbiorem terminów zalecanych T_{0j} , tzw. deskryptorów, i pozostałych terminów pomocniczych - askryptorów.

Określona jest zwrotna, symetryczna i przechodnia relacja synonimi s_j oraz przeciwzwrotna, przechodnia, a więc i przeciwsymetryczna relacja uogólnienia R_j .

W wyniku klasyfikacji powstaje opis dokumentu $d \in D_j$

$$t(d) = \{t_1, t_2, \dots, t_k\}$$

spełniający warunek

$$t' \neq t \wedge t' R t \Rightarrow t' \notin t_j(d) \vee t'' \notin t(d).$$

Kwerenda $q \in Q_j$ ma taką samą postać jak opis dokumentu. Zbiór opisów dokumentów i kwerend jest częściowo uporządkowany przez relację zawierania $\succ_j$ zdefiniowaną następująco

$$t(d_1) \succ_j t_j(d_2) \Leftrightarrow (\forall t' \in t_j(d_1)) (\exists t'' \in t_j(d_2)) (t' = t'' \vee t' R_j t'')$$

Odpowiedzią na kwerendę $q \in Q_j$ nazywamy

$$f_j(q) = \{d: d \in D_j \wedge q \succ_j t_j(d)\}.$$

Przy tak zdefiniowanej odpowiedzi system S_j jest inkluzyjny [56], tzn.

$$\{q_1, q_2\} \subset Q_j \wedge q_1 \succ_j q_2 \Rightarrow f_j(q_2) \subset f_j(q_1).$$

Definicja 12

Rozproszonym systemem wyszukiwania informacji opartym na globalnym teaurusie [56] nazywać będziemy system $S = (T, D, Q, f)$ zbudowany na bazie n lokalnych systemów $S_j = (T_j, D_j, Q_j, f_j)$ i spełniający warunki:

(i) T jest globalnym teaurusem z relacją uogólnienia R określoną na zbiorze deskryptorów T_{0j} ,

$$T = \bigcup_{j=1}^n T_j, T_0 = \bigcup_{j=1}^n T_{0j}, R_j = R \cap (T_{0j} \times T_{0j}),$$

(ii) D jest zbiorem dokumentów, $D = \bigcup_{j=1}^n D_j$,

(iii) Q jest zbiorem kwerend $Q \supset \bigcup_{j=1}^n Q_j$ oraz $\succ_j = \succ \cap (Q_j \times Q_j)$,

(iv) $f: Q \rightarrow 2^D$ jest odwzorowaniem przyporządkowującym każdej kwerendzie zbiór dokumentów, $f(q) = \{d: d \in D \wedge q \succ t(d)\}$.

■

Relacja zawierania się $\succ$ jest zdefiniowana w zbiorze opisów dokumentów tak samo jak w zbiorze kwerend.

Definicja 13

Dla każdego j , ($1 \leq j \leq n$) lokalny system $S_j = (T_j, D_j, Q_j, f_j)$ jest podsystemem rozproszonego systemu $S = (T, D, Q, f)$, jeśli

(i) $T_j \subset T \wedge R_j = R \cap (T_{0j} \times T_{0j})$,

(ii) $D_j \subset D$,

(iii) $Q_j \subset Q \wedge \succ_j = \succ \cap (Q_j \times Q_j)$,

(iv) $f_j(q) = f(q) \cap D_j$ dla $q \in Q_j$.

■

Dowolna globalna kwerenda q należy do zbioru Q_j , jeśli

$$\forall_{t_i \in q} X_i = \{t \in T: t_i = t \vee t_i R t\} \subset T_j.$$

Aby rozproszony system był dobrze określony, musi być również spełniony warunek zgodności:

$$d \in D_j \cap D_k \wedge t \in T_j \cap T_k \wedge tet_j(d) \Rightarrow tet_k(d).$$

Spróbujmy teraz odpowiedź na globalną kwerendę wyrazić jako wynik określonych operacji na odpowiedziach z lokalnych systemów. Niech $f_j^*(t)$ będzie odpowiedzią na kwerendę składającą się z pojedynczego deskryptora

$$f_j^*(t) = \{d: d \in D_j \wedge tet_j(d)\} \quad (20)$$

przy założeniu, że jeśli $t \notin T_j$, to $f_j^*(t) = \emptyset$.

Definicja 14

Niech

$$q = \{t_1, t_2, \dots, t_k\} \quad (22)$$

będzie globalną kwerendą. Oznaczamy przez X_i zbiór wszystkich deskryptorów ze zbioru T bardziej szczegółowych niż deskryptor t_i ,

$$X_i = \{t: t \in T \wedge (t_i = t \vee t_i R t)\}.$$

Odpowiedź na dowolną kwerendę (21) ma postać

$$f(q) = \bigcap_{i=1}^k \bigcup_{t \in X_i} \bigcup_{j=1}^n f_j^*(t)$$

■

Rozproszony system oparty na teaurusie jest inkluzyjny, pełny dowód podano w [56]. Pokażemy teraz sposób wykorzystania istniejących lokalnych list inwersyjnych w celu poszukania odpowiedzi w rozproszonym systemie. W każdym lokalnym podsystemie S_j lista inwersyjna $L_j(t)$ jest zbiorem dokumentów, które zawierają deskryptor t w swoim opisie

$$\forall_{t \in T_j} L_j(t) = \{d: d \in D_j \wedge tet_j(d)\}.$$

Katwo zauważyć, że postać odpowiedzi częściowej, podanej w związku (21), pokrywa się z definicją listy inwersyjnej

$$\forall_{t \in T_j} L_j(t) = f_j^*(t).$$

Zgodnie z definicją 12, odpowiedź na kwerendę postaci (22) możemy przedstawić jako

$$f(q) = \bigcap_{i=1}^k \left[\bigcup_{j=1}^n \bigcup_{x \in X_i} L_j(t) \right] \quad (23)$$

przy założeniu, że $L_j(t) = \emptyset$ dla $t \in T_j$.

Tak zdefiniowana operacja na listach inwersyjnych również zachowuje własność inkluzyjności [51], [52].

5.2. Rozproszony system wyszukiwania informacji oparty na modelach wagowych

W każdym lokalnym systemie $S_j = (T_j, D_j, \theta_j, f_j)$ [48] tezaurus T_j jest zbiorem terminów zalecanych – deskryptorów T_{0j} i pozostałych terminów pomocniczych – askryptorów. Określona jest relacja synonimii ϕ_j oraz relacja równości R_j taka, że $t_1 R_j t_2 \Leftrightarrow t_1 = t_2$.

W wyniku klasyfikacji powstaje opis dokumentu $d \in D_j$

$$t_j(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\} \quad (24)$$

spełniający warunki

$$(i) \quad \forall_{1 \leq i \leq k} t_i \in T_j \wedge 0 \leq w_i \leq 1,$$

$$(ii) \quad \sum_{i=1}^k w_i \leq 1.$$

Kwerenda $q \in Q_j$ ma taką samą postać jak opis dokumentu, suma wag w kwerendzie jest równa jedności. Zbiór opisów dokumentów i kwerend jest częściowo uporządkowany przez relację zawierania $\succ_{jc}$ z dokładnością c zdefiniowaną następująco

$$t_j(d_1) \succ_{jc} t_j(d_2) \Leftrightarrow \begin{aligned} & \Leftrightarrow \forall (t', w') \in t_j(d_1) \exists (t'', w'') \in t_j(d_2) \\ & t' R_j t'' \wedge w' \leq w'' \end{aligned} \quad (25)$$

Określony jest stopień zawierania się kwerend w opisach dokumentów spełniających warunek (25). Dla opisów dokumentów postaci (4), (5)

stopień zawierania μ ma postać

$$\mu(t_j(d_1), t_j(d_2)) = \sum_{k \in N_i} \min(w_{1k}, w_{2k}),$$

gdzie $N_i = \{k: t_{1i} = t_{2m} \wedge 1 \leq m \leq k_2\}$ dla $1 \leq i \leq k$.

Odpowiedzią na kwerendę $q \in Q_j$ z dokładnością c ($0 \leq c \leq 1$) nazywamy

$$f_j(q, c) = \{(d, v): d \in D_j \wedge q \succ_{jc} t_j(d) \wedge v = \mu(q, t_j(d))\}.$$

Podstawowe własności odpowiedzi można wyrazić w postaci twierdzenia.

Twierdzenie 16

- (i) Jeśli $(d, v) \in f_j(q, c)$, to $v \geq c$.
- (ii) Jeśli $c_1 \geq c_2$, to $f_j(q, c_1) \subset f_j(q, c_2)$.
- (iii) Własność inkluzyjności systemu S_j jest zachowana, tzn. prawdziwa jest implikacja

$$\{q_1, q_2\} \subset Q_j \wedge q_1 \succ_{jc} q_2 \wedge c_1 \leq c_2 \Rightarrow f_j(q_1, c_1) \supset f_j(q_2, c_2).$$

Dowód tego twierdzenia oraz inne własności systemów opartych na modelach wagowych podano w [48].

■

Na bazie n lokalnych systemów S_j wyszukiwania informacji można zbudować rozproszony system $S = (T, D, Q, f)$ [48] spełniający warunki:

- (i) T jest globalnym tezaurem z relacją równości R określoną na zbiorze deskryptorów T_{Oj} ,

$$T = \bigcup_{j=1}^n T_j, \quad T_O = \bigcup_{j=1}^n T_{Oj}, \quad R_j = R \cap (T_{Oj} \times T_{Oj}),$$

- (ii) D jest zbiorem dokumentów, $D = \bigcup_{j=1}^n D_j$,

- (iii) Q jest zbiorem kwerend; $Q \supset \bigcup_{j=1}^n Q_j$ oraz $\succ_{jc} = \succ_c \cap (Q_j * Q_j)$,

- (iv) $f: Q \times [0, 1] \rightarrow 2^{D \times [0, 1]}$ jest odwzorowaniem przyporządkowującym każdej parze (kwerenda q , żądana dokładność c odpowiedzi) zbiór par (dokument, miara podobieństwa kwerendy do opisu dokumentu)

$$f(q, c) = \{(d, v): d \in D \wedge q \succ_c t(d) \wedge v = \mu(q, t(d))\}.$$

Dowolna globalna kwerenda q należy do zbioru Q_j , jeśli każdy deskryptor z kwerendy q należy do zbioru T_j . Aby rozproszony system był dobrze określony, musi być spełniony warunek zgodności (15). Odpowiedź na globalną kwerendę można przedstawić jako wynik określonych operacji na odpowiedziach z lokalnych systemów.

Niech $f_j^*(q, c)$ będzie odpowiedzią z dokładnością c na kwerendę $q = \{(t, w)\}$ składającą się z pojedynczego deskryptora t z wagą w

$$f_j^*(q, c) = \{(d, v) : d \in D_j \wedge (t, v) \in t_j(d) \wedge wc \leq v\}$$

przy założeniu, że $f_j^*(q, c) = \emptyset$ dla $q \notin Q_j$.

W rozproszonym systemie S odpowiedź na tę kwerendę jest sumą lokalnych odpowiedzi

$$f^*(q, c) = \bigcup_{j=1}^n f_j^*(q, c). \quad (26)$$

Definicja 15

Niech $q_1 = \{(t_1, w_1)\}$, $q_2 = \{(t_2, w_2)\}$ będą kwerendami.

$$f_j^*(q_1, c) \cap f_j^*(q_2, c) = \{(d, v) : (d, v_1) \in f_j^*(q_1, c) \wedge v_1 \geq cw_1 \wedge \\ (d, v_2) \in f_j^*(q_2, c) \wedge v_2 \geq cw_2 \wedge v = \min(w_1, v_1) + \min(w_2, v_2)\}.$$

■

W rozproszonym systemie S odpowiedź na dowolną kwerendę

$$q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\} \in Q \quad (27)$$

można przedstawić jako

$$f(q, c) = \bigcap_{j=1}^k f_j^*(q_j, c), \text{ gdzie } q_j = \{(t_j, w_j)\}.$$

Lokalne listy inwersyjne mogą być wykorzystane do poszukiwania globalnych odpowiedzi. W każdym podsystemie S_j lista inwersyjna $L_j(t)$ jest zbiorem dokumentów z wagami, które zawierają deskryptor t i jego wagę w swoim opisie

$$\forall_{t \in T_j} L_j(t) = \{(d, v) : d \in D_j \wedge (t, v) \in t_j(d)\}.$$

Łatwo zauważyć, że postać odpowiedzi częściowej (26) pokrywa się z definicją listy inwersyjnej

$$\forall_{t \in T_j} L_j(t) = f_j^*(t).$$

Definicja 16

$$(i) \quad (d, v) \in L(t) \Big|_a \Leftrightarrow (d, v) \in L(t) \wedge v \geq a.$$

(ii) Niech $q_1 = \{(t_1, w_1)\}$, $q_2 = \{(t_2, w_2)\}$ będą kwerendami.

$$(d, v) \in L(t_1) \Big|_{w_1 c} \cap L(t_2) \Big|_{w_2 c} \Leftrightarrow (d_1, v_1) \in L(t_1) \Big|_{w_1 c} \wedge \\ \wedge (d, v_2) \in L(t_2) \Big|_{w_2 c} \wedge v = \min(v_1, w_1) + \min(v_2, w_2).$$

Zgodnie z definicją 14, odpowiedź na kwerendę (27) można przedstawić jako wynik określonych operacji na lokalnych listach inwersyjnych

$$f(q, c) = \bigcap_{i=1}^k \bigcup_{j=1}^n \left(L_j(t) \Big|_{cw_i} \right).$$

5.3. Rozproszony system wyszukiwania informacji oparty na zbiorach terminów

W każdym lokalnym systemie $S_j = (T_j, D_j, Q_j, f_j)$ zbiór T_j zawiera terminy zalecane (deskryptory) tworzące podzbiór T_{0j} i pozostałe terminy pomocnicze - askryptory.

Określona jest relacja synonimii oraz relacja równości R_j taka, że

$$t_1 R t_2 \Leftrightarrow t_1 = t_2.$$

W wyniku klasyfikacji powstaje opis dokumentu $d \in D_j$

$$t_j(d) = \{t_1, t_2, \dots, t_k\}. \quad (28)$$

Kwerenda $q \in Q_j$ ma taką samą postać jak opis dokumentu. Zbiór opisów dokumentów i kwerend jest częściowo uporządkowany przez relację zawierania $\succ_j$ zdefiniowaną następująco:

$$t_j(d_1) \succ_j t_j(d_2) \Leftrightarrow \forall t' \in t_j(d_1) \exists t'' \in t_j(d_2) t' R_j t''.$$

Odpowiedzią na kwerendę $q \in Q_j$ nazywamy

$$f_j(q, c) = \{d: d \in D_j \wedge q \succ_j t_j(d)\}.$$

Przy tak zdefiniowanej odpowiedzi system S_j jest inkluzyjny, tzn.

$$\{q_1, q_2\} \subset Q_j \wedge q_1 \succ_j q_2 \Rightarrow f_j(q_2) \subset f_j(q_1).$$

Definicja 17

Rozproszonym systemem wyszukiwania informacji opartym na zbiorach terminów [49] nazywać będziemy system $S = (T, D, Q, f)$ zbudowany na bazie n lokalnych systemów $S_j = (T_j, D_j, Q_j, f_j)$ i spełniający warunki:

(i) T jest globalnym zbiorem z relacją równości R określoną na zbiorze deskryptorów T_{0j} ,

$$T = \bigcup_{j=1}^n T_j, \quad T_0 = \bigcup_{j=1}^n T_{0j}, \quad R_j = R \cap (T_{0j} \times T_{0j}),$$

(ii) D jest zbiorem dokumentów, $D = \bigcup_{j=1}^n D_j$,

(iii) Q jest zbiorem kwerend, $Q \supset \bigcup_{j=1}^n Q_j$ oraz $\succ_j = \succ \cap (Q_j \times Q_j)$,

(iv) $f: Q \rightarrow 2^D$ jest odwzorowaniem przyporządkowującym każdej kwerendzie zbiór dokumentów, $f(q) = \{d: d \in D \wedge q \succ t(d)\}$.

■

Dowolna globalna kwerenda q należy do zbioru Q_j , jeśli każdy deskryptor z kwerendy q należy do zbioru T_j . Aby system S był dobrze określony, musi być spełniony warunek zgodności:

$$d \in D_j \cap D_k \wedge t \in T_j \cap T_k \wedge t \prec_j(d) \Rightarrow t \prec_k(d).$$

Odpowiedź na globalną kwerendę można przedstawić jako wynik pewnych operacji odpowiedzi z lokalnych podsystemów.

Niech $f_j^*(q)$ będzie odpowiedzią na kwerendę $q = \{t\}$ składającą się z pojedynczego deskryptora,

$$f_j^*(q) = \{d: d \in D_j \wedge t \prec_j(d)\}$$

przy założeniu, że $f_j^*(q) = \emptyset$ dla $q \notin Q_j$.

W rozproszonym systemie S odpowiedź na tę samą kwerendę jest sumą lokalnych odpowiedzi $f^*(q) = \bigcup_{j=1}^n f_j^*(q)$.

Odpowiedź na dowolną kwerendę $q = \{t_1, \dots, t_k\} \in Q$ można przedstawić jako

$$f(q) = \bigcap_{i=1}^k f^*(q_i), \text{ gdzie } q_i = \{t_i\}.$$

Rozproszony system oparty na zbiorach terminów jest inkluzyjny [49]. Istniejące w lokalnych podsystemach listy inwersyjne mogą być wykorzystane do poszukiwania odpowiedzi na globalne kwerendy. W każdym lokalnym podsystemie S_j lista inwersyjna $L_j(t)$ jest zbiorem dokumentów, które zawierają deskryptor t w swoim opisie

$$\forall_{t \in T_j} L_j(t) = \{d : d \in D_j \wedge t \in t_j(d)\}.$$

Odpowiedź na dowolną kwerendę $q = \{t_1, t_2, \dots, t_k\}$ można przedstawić jako wynik pewnych operacji na listach inwersyjnych

$$f(q) = \bigcap_{i=1}^k \bigcup_{j=1}^n L_j(t_i)$$

przy założeniu, że $L_j(t) = \emptyset$ dla $t \notin T_j$.

6. ROZPROSZONY SYSTEM WYSZUKIWANIA INFORMACJI OPARTY NA HETEROGENICZNYCH PODSYSTEMACH

Założmy, że istnieją systemy wyszukiwania informacji oparte na modelach rozpatrywanych w poprzednich rozdziałach monografii. Różnorodne lokalne systemy możemy również połączyć w sieć, tworząc w ten sposób jeden rozproszony system wyszukiwania informacji oparty na heterogenicznych podsystemach.

Niech SHH będzie systemem opartym na teźaurusie z wagami, zdefiniowanym w rozdziale 2, SH – systemem opartym na teźaurusie (rozdz.5.1), $\bar{SH}$ – systemem opartym na modelu wagowym (rozdz.5.2.), ST – systemem opartym na zbiorach terminów (rozdz.5.3.).

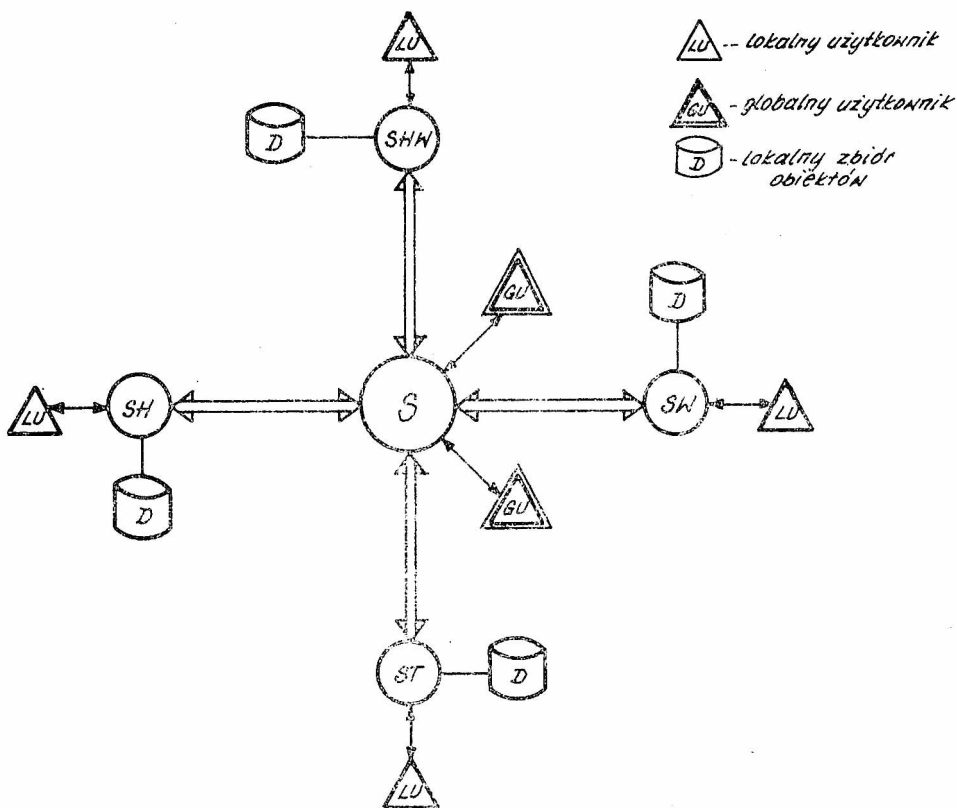
System S będący złożeniem różnorodnych systemów, spełniających warunki podane w definicji 7, nazywać będziemy rozproszonym systemem zbudowanym na bazie heterogenicznych podsystemów typów SHH , SH , $\bar{SH}$, ST (rys. 6).

Głównym problemem projektantów rozproszonych systemów jest konieczność podania metody, która pozwala globalną kwerendę zamienić na częściowe kwerendy w językach lokalnych, takich że operacje na częściowych odpowiedziach pozwolą uzyskać odpowiedź na globalną kwerendę. Wynika to stąd, że globalny użytkownik rozproszonego systemu widzi system jako całość i nie chce wiedzieć, czy ten system jest rozproszony czy też nie. Faktycznie taki system istnieje tylko na poziomie logicznym, fizycznie składa się on z kilku połączonych ze sobą podsystemów, w których przechowywane są informacje. Dlatego też uzyskanie globalnej odpowiedzi polega na złożeniu częściowych odpowiedzi z lokalnych podsystemów.

W procesie dekompozycji globalna kwerenda q dzielona jest na kwerendy częściowe q_i , które po usunięciu deskryptorów nie należących do lokalnych teźaurusów T_i są kierowane do odpowiednich podsystemów. Globalna odpowiedź f powstaje w wyniku operacji na częściowych odpowiedziach f_j^* , otrzymanych z lokalnych podsystemów zgodnie z (19)

$$f(q, c) = \bigcap_{i=1}^k \{ (d, v_i) : (d, v_i) \in \bigcup_{t \in X_i} \bigcup_{j=1}^n f_j^*(t) \wedge v_i \geq w_i c \},$$

gdzie $(t_i, w_i) \in q$ dla $1 \leq i \leq k$.



Rys. 6. Architektura rozproszonego systemu wyszukiwania informacji
 Fig. 6. Architecture of distributed information retrieval system

Postać częściowej odpowiedzi f_j^* zależy od typu podsystemu. Podamy teraz warunki, jakie muszą spełniać podsystemy, aby mogły współpracować z rozproszonym systemem S . Globalny system S wyszukiwania informacji oparty będzie na teźaurusie z wagami, wszystkie jego własności mogą być wyrażane w lokalnych podsystemach.

6.1. Połączenie typu S-ST

Aby podsystem $S_1 = (D_1, T_1, Q_1, f_1)$, typu ST, oparty na zbiorze terminów mógł być włączony do rozproszonego systemu $S = (D, T, Q, f)$, muszą być spełnione, zgodnie z definicją 7, warunki:

(i) Do zbioru deskryptorów $T_{10} = \{t_1, t_2, \dots, t_n\} \subset T_1$ mogą należeć tylko te deskryptory ze zbioru $T_0 \subset T$, które nie są w relacji łagodnienia, tzn. nie zachodzi $t_i R t_k$ dla $i, k \in \{1, 2, \dots, n\}$. Moc zbioru $T_{10} \subset T_1$ jest mniejsza lub równa mocy zbioru wszystkich deskryptorów należących do $T_0 \subset T$ parami nieporównywalnych w sensie relacji R , tj.

$$\text{card}(T_{10}) \leq \text{card}((T_0 \times T_0) \setminus R)$$

(ii) Dla każdego $t \in T_{10}$ w podsystemie S_1 istnieje pomocniczy zbiór $I_1(t)$ zawierający informacje o wagach deskryptora t występującego w globalnych opisach dokumentów należących do zbioru D_1 . Tworząc globalne opisy $t(d) = \bigcup_{j=1}^n t_j(d)$ dla dokumentów $d \in D_1$, sprawimy, że niektóre opisy mogą zawierać deskryptory z zerowymi wagami. Takie przypadki występować będą tylko wtedy, gdy deskryptory te będą należeć do podsystemów niewagowych. Zbiór $I_1(t)$ przyjmuje postać:

$$I_1(t) = \{v: (t, v) \in t(d) \wedge v \neq 0 \wedge d \in D_1\} \cup \\ \cup \left\{ (1 - \sum_{v \neq 0} v) / \text{card } t(d): (t', v') \in t(d) \wedge v' = 0 \wedge d \in D_1 \right\}.$$

Zbiór ten będzie wykorzystywany tylko podczas przygotowywania częściowej odpowiedzi dla globalnego systemu.

W lokalnym podsystemie S_1 zbiór wag $I_1(t)$ ma dokładnie taką samą strukturę jak lista inwersyjna $L_1(t)$, tzn. jeśli dokument d występuje na i -tym miejscu listy $L_1(t)$ oraz $(t, v) \in t(d)$, to waga v również wystąpi na i -tym miejscu w zbiorze $I_1(t)$ odpowiadającym temu deskryptorowi.

Częściowa odpowiedź otrzymana z podsystemu S_1 ma postać

$$f_1^*(t) = \{(d, v): d \in D_1 \wedge t \in t_1(d) \wedge v \in I_1(t)\}.$$

Łatwo można zauważyć, że w przypadku wykorzystania metody list inwersyjnych odpowiedź częściową można przedstawić jako

$$f_1^*(t) = \{(d, v): d \in L_1(t) \wedge v \in I_1(t)\}.$$

Przykład 10

Niech $S_1 = (T_1, D_1, Q_1, f_1)$ będzie danym podsystemem wyszukiwania, w którym $T_1 = \{t_4, t_6, t_7, t_{11}, t_{12}\}$ jest zbiorem deskryptorów, a opisy dokumentów mają postać:

$$t_1(d_1) = \{t_6\}$$

$$t_1(d_2) = \{t_4, t_{11}\}$$

$$t_1(d_4) = \{t_6, t_7, t_{11}\}$$

$$t_1(d_5) = \{t_6, t_{12}\}$$

$$t_1(d_7) = \{t_4, t_7, t_{12}\}$$

$$t_1(d_8) = \{t_6\}.$$

Listy inwersyjne dla deskryptorów należących do T_1 będą miały postać:

$$L_1(t_4) = \{d_2, d_7\}$$

$$L_1(t_6) = \{d_1, d_4, d_5, d_8\}$$

$$L_1(t_7) = \{d_4, d_7\}$$

$$L_1(t_{11}) = \{d_2, d_4\}$$

$$L_1(t_{12}) = \{d_5, d_7\}.$$

Aby lokalne listy inwersyjne mogły być udostępnione globalnym użytkownikom, należy utworzyć pomocnicze zbiory uporządkowane zawierające wagi deskryptorów, z jakimi występują one w globalnych opisach dokumentów (podanych w rozdziale 6.5)

$$I_1(t_4) = \{0.3, 0.3\}$$

$$I_1(t_6) = \{0.4, 0.2, 0.4, 0.3\}$$

$$I_1(t_7) = \{0.2, 0.3\}$$

$$I_1(t_{11}) = \{0.3, 0.3\}$$

$$I_1(t_{12}) = \{0.2, 0.4\}.$$

Kolejne elementy w zbiorach $I_1(t)$ odpowiadają kolejnym elementom z list inwersyjnych $L_1(t)$.

6.2. Połączenie typu S-SH

System $S_2 = (D_2, T_2, Q_2, f_2)$ typu SH, oparty na modelu wagowym, będzie podsystem rozproszonego systemu $S = (D, T, Q, f)$, gdy do zbioru deskryptorów T_{20} będą należały te deskryptory ze zbioru $T_0 \subset T$, które nie są w relacji uogólnienia, tzn. $t_i \sim R t_k$ dla $(t_i, t_k \in T_{02})$.

Prawdziwa jest nierówność

$$\text{card}(T_{20}) \leq \text{card}((T_0 \times \emptyset) \setminus R).$$

Częściowa odpowiedź z podsystemu S_2 przyjmie postać

$$f_j^*(t) = \{(d, v) : d \in D_2 \wedge (t, v) \in t_2(d)\}$$

i w przypadku wykorzystania metody list inwersyjnych częściowa odpowiedź jest równa odpowiedniej liście inwersyjnej, tj.

$$f_j^*(t) = L_2(t).$$

Przykład 11

Niech $S_2 = (D_2, T_2, Q_2, f_2)$ będzie systemem wyszukiwania informacji opartym na modelu wagowym, w którym $T_2 = \{t_7, t_{10}, t_{11}, t_{12}\}$, a opisy dokumentów mają postać:

$$t_2(d_2) = \{(t_{11}, 0.3)\}$$

$$t_2(d_3) = \{(t_{10}, 0.3)\}$$

$$t_2(d_4) = \{(t_{10}, 0.3), (t_{11}, 0.3), (t_7, 0.2)\}$$

$$t_2(d_5) = \{(t_{12}, 0.2)\}$$

$$t_2(d_7) = \{(t_{12}, 0.4), (t_7, 0.3)\}$$

$$t_2(d_8) = \{(t_{10}, 0.4)\}.$$

Listy inwersyjne dla deskryptorów należących do T_2 będą miały postać:

$$L_2(t_7) = \{(d_4, 0.2), (d_7, 0.3)\}$$

$$L_2(t_{10}) = \{(d_3, 0.3), (d_4, 0.3), (d_8, 0.4)\}$$

$$L_2(t_{11}) = \{(d_2, 0.3), (d_4, 0.3)\}$$

$$L_2(t_{12}) = \{(d_5, 0.2), (d_7, 0.4)\}.$$

6.3. Połączenie typu S-SH

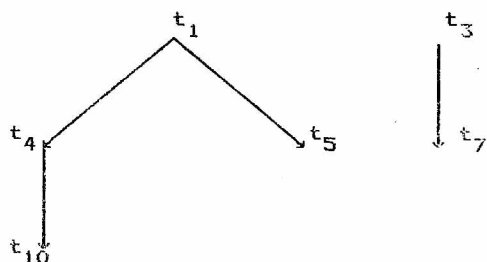
Podsystem $S_3 = (D_3, T_3, Q_3, f_3)$ typu SH, oparty na tezaurusie, będzie mógł być włączony do rozproszonego systemu $S = (D, T, Q, f)$, gdy, zgodnie z definicją 7, w podsystemie S_3 będzie istniał dla każdego deskryptora T_{30} pomocniczy zbiór $I_3(t) = \{v: (t, v) \in t(d) \wedge d \in D_1\}$ zawierający informacje o wagach deskryptorów występujących w globalnym systemie S. Zbiór ten będzie wykorzystany tylko podczas przygotowania częściowej odpowiedzi dla globalnego systemu. Fizyczna postać uporządkowanego zbioru wag $I_3(t)$ będzie dokładnie taka sama jak realizacja metody wyszukiwania informacji w lokalnym podsystemie S_3 typu SH.

Odpowiedź częściowa z podsystemu S_3 ma postać:

$$f_3^*(t) = \{(d, v): d \in D_3 \wedge t \in t_3(d) \wedge v \in I_3(t)\}.$$

Przykład 12

Niech $S_3 = (D_3, T_3, Q_3, f_3)$ typu *SH* będzie systemem wyszukiwania informacji opartym na tezaurusie, w którym $T_3 = \{t_1, t_3, t_4, t_5, t_7, t_{10}\}$ jest tezaurusem. Relację uogólniania R_3 zachodzącą pomiędzy deskryptorami przedstawmy graficznie (rys. 7)

Rys. 7. Postać lokalnego tezaurya T_3 Fig. 7. Structure of the local thesaurus T

oraz niech opisy dokumentów mają postać:

$$t_3(d_1) = \{t_5\}$$

$$t_3(d_2) = \{t_4\}$$

$$t_3(d_3) = \{t_{10}, t_5, t_3\}$$

$$t_3(d_4) = \{t_{10}, t_7\}$$

$$t_3(d_6) = \{t_5, t_3\}$$

$$t_3(d_7) = \{t_4, t_7\}$$

$$t_3(d_8) = \{t_{10}, t_5\}$$

Listy inwersyjne dla poszczególnych deskryptorów będą miały postać:

$$L_3(t_1) = \emptyset$$

$$L_3(t_3) = \{d_3, d_6\}$$

$$L_3(t_4) = \{d_2, d_7\}$$

$$L_3(t_5) = \{d_1, d_3, d_6, d_8\}$$

$$L_3(t_7) = \{d_4, d_7\}$$

$$L_3(t_{10}) = \{d_3, d_4, d_8\}$$

Aby lokalne listy inwersyjne mogły być udostępnione globalnym użytkownikom, należy utworzyć pomocnicze zbiory uporządkowane zawierające wagi deskryptorów, z jakimi występują one w globalnych opisach dokumentów (podanych w rozdziale 6.5)

$$I_3(t_1) = \emptyset$$

$$I_3(t_3) = \{0.2, 0.4\}$$

$$I_3(t_4) = \{0.3, 0.3\}$$

$$I_3(t_5) = \{0.2, 0.3, 0.6, 0.3\}$$

$$I_3(t_7) = \{0.2, 0.2\}$$

$$I_3(t_{10}) = \{0.3, 0.3, 0.4\}$$

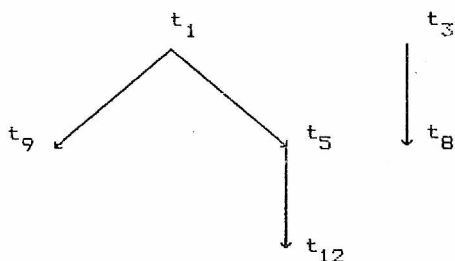
Kolejne elementy w zbiorach $I_3(t)$ odpowiadają kolejnym dokumentom z list inwersyjnych $L_3(t)$.

6.4. Połączenie typu S-SHH

Podsystem $S_4 = (D_4, T_4, Q_4, f_4)$ oparty na tezaurysie z wagami może być włączony do rozproszonego systemu z warunkami podanymi szczegółowo w rozdziale 3.

Przykład 13

Niech $S_4 = (D_4, T_4, Q_4, f_4)$ będzie system wyszukiwania informacji opartym na tezaurysie z wagami, w którym $T_4 = \{t_1, t_3, t_5, t_8, t_9, t_{12}\}$ jest tezaurem. Relację uogólniania zachodzącą pomiędzy deskryptorami przedstawmy graficznie (rys. 8)



Rys. 8. Postać lokalnego tezaurusa T_4

Fig. 8. Structure of the local thesaurus T_4

Opisy dokumentów niech będą postaci:

$$t_4(d_1) = \{(t_5, 0.2), (t_8, 0.4)\}$$

$$t_4(d_2) = \{(t_8, 0.4)\}$$

$$t_4(d_3) = \{ (t_9, 0.2), (t_5, 0.3), (t_3, 0.2) \}$$

$$t_4(d_5) = \{ (t_9, 0.5), (t_{12}, 0.2), (t_8, 0.1) \}$$

$$t_4(d_6) = \{ (t_5, 0.6), (t_3, 0.4) \}$$

$$t_4(d_7) = \{ (t_{12}, 0.4) \}$$

$$t_4(d_8) = \{ (t_5, 0.3) \}.$$

Listy inwersyjne dla poszczególnych deskryptorów w podsystemie S_4 są postaci:

$$L_4(t_1) = \emptyset$$

$$L_4(t_3) = \{ (d_3, 0.2), (d_6, 0.4) \}$$

$$L_4(t_5) = \{ (d_1, 0.2), (d_3, 0.3), (d_6, 0.6), (d_8, 0.3) \}$$

$$L_4(t_8) = \{ (d_1, 0.4), (d_2, 0.4), (d_5, 0.1) \}$$

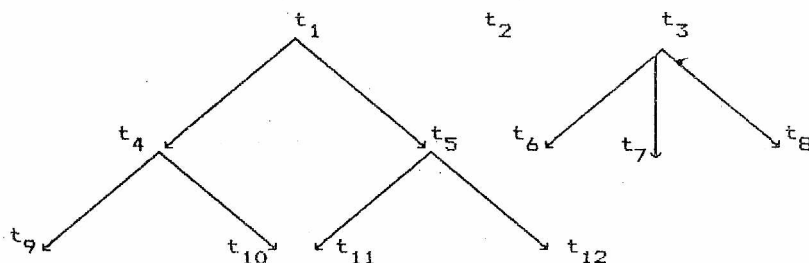
$$L_4(t_9) = \{ (d_3, 0.2), (d_5, 0.5) \}$$

$$L_4(t_{12}) = \{ (d_5, 0.2), (d_7, 0.4) \}.$$

6.5. Przykład rozproszonego systemu opartego na heterogenicznych podsystemach

Założmy, że mamy 4 różnorodne podsystemy typu SHH , SH , SH , ST (rys.6), zdefiniowane w poprzednich rozdziałach. Zilustrujemy przykładem sposób tworzenia globalnej odpowiedzi na podstawie częściowych odpowiedzi f_j^* otrzymanych z lokalnych podsystemów. Założmy ponadto, że tezaury w lokalnych podsystemach S_1, S_2, S_3, S_4 mają taką samą postać jak w przykładach 10-13.

Na bazie tych podsystemów można zbudować rozproszony system wyszukiwania informacji, w którym $T = \{t_1, t_2, t_3, \dots, t_{12}\}$ jest globalnym tezaurem z relacją uogólnienia przedstawioną graficznie (rys.9)



Rys. 9. Postać globalnego tezaury T
Fig. 9. Structure of the global thesaurus T

oraz globalnymi opisami dokumentów postaci:

$$\begin{aligned}
 t(d_1) &= \{ (t_5, 0.2), (t_6, 0.4), (t_8, 0.4) \} \\
 t(d_2) &= \{ (t_4, 0.3), (t_{11}, 0.3), (t_8, 0.4) \} \\
 t(d_3) &= \{ (t_9, 0.2), (t_{10}, 0.3), (t_5, 0.3), (t_3, 0.2) \} \\
 t(d_4) &= \{ (t_{10}, 0.3), (t_{11}, 0.3), (t_6, 0.2), (t_7, 0.2) \} \\
 t(d_5) &= \{ (t_9, 0.5), (t_{12}, 0.2), (t_6, 0.4), (t_8, 0.1) \} \\
 t(d_6) &= \{ (t_5, 0.6), (t_3, 0.4) \} \\
 t(d_7) &= \{ (t_4, 0.3), (t_{12}, 0.4), (t_7, 0.3) \} \\
 t(d_8) &= \{ (t_{10}, 0.4), (t_5, 0.3), (t_6, 0.3) \}.
 \end{aligned}$$

Niech $q = \{ (t_4, 0.3), (t_5, 0.3), (t_3, 0.4) \}$ będzie kwerendą globalnego użytkownika wymagającego odpowiedzi z dokładnością $c=0.5$. Aby lokalne listy inwersyjne z podsystemów SH i ST (opartych na modelach niewagowych) mogły być wykorzystywane w rozproszonym systemie, należy w poszczególnych listach przyporządkować dokumentom wagi należące do uporządkowanego zbioru $I(t)$, tzn. listy te przyjmują postać:

$$L_j^*(t) = \{ (d, v) : d \in L_j(t) \wedge v \in I_j(t) \}.$$

Zgodnie z wyrażeniem (20), dla $n=4$ globalna odpowiedź w heterogenicznym systemie ma postać:

$$\begin{aligned}
 f(q, c) &= \bigcap_{i=1}^n \left[\bigcup_{t \in X} \bigcup_{j=1}^n L_j(t_i) \right] \Big|_{cw_i} = \\
 &= \left[\bigcup_{j=1}^4 L_j(t_4) \bar{\cup} \bigcup_{j=1}^4 L_j(t_9) \bar{\cup} \bigcup_{j=1}^4 L_j(t_{10}) \right] \Big|_{0.5*0.3} \cap \\
 &\quad \cap \left[\bigcup_{j=1}^4 L_j(t_5) \bar{\cup} \bigcup_{j=1}^4 L_j(t_{11}) \bar{\cup} \bigcup_{j=1}^4 L_j(t_{12}) \right] \Big|_{0.5*0.3} \cap \\
 &\quad \cap \left[\bigcup_{j=1}^4 L_j(t_3) \bar{\cup} \bigcup_{j=1}^4 L_j(t_6) \bar{\cup} \bigcup_{j=1}^4 L_j(t_7) \bar{\cup} \bigcup_{j=1}^4 L_j(t_8) \right] \Big|_{0.5*0.4} = \\
 &= \left[\{ (d_2, 0.3), (d_7, 0.3) \} \bar{\cup} \{ (d_3, 0.2), (d_5, 0.5) \} \bar{\cup} \right. \\
 &\quad \left. \bar{\cup} \{ (d_3, 0.3), (d_4, 0.3), (d_8, 0.4) \} \right] \Big|_{0.5*0.3} \\
 &\quad \cap \left[\{ (d_1, 0.2), (d_3, 0.3), (d_6, 0.6), (d_8, 0.3) \} \bar{\cup} \right. \\
 &\quad \left. \bar{\cup} \{ (d_2, 0.3), (d_4, 0.3) \} \bar{\cup} \{ (d_5, 0.2), (d_7, 0.4) \} \right] \Big|_{0.5*0.3} \\
 &\quad \cap \left[\{ (d_3, 0.2), (d_6, 0.4) \} \bar{\cup} \{ (d_1, 0.4), (d_5, 0.4), (d_8, 0.3) \} \bar{\cup} \right.
 \end{aligned}$$

$$\begin{aligned}
& \bar{U} \{ (d_4, 0.2), (d_7, 0.3) \} \bar{U} \{ (d_1, 0.4), (d_2, 0.4), (d_5, 0.1) \} \Big|_{0.5 * 0.4} \\
&= \{ (d_2, 0.3), (d_7, 0.3), (d_3, 0.5), (d_5, 0.5), (d_4, 0.3), (d_8, 0.4) \} \Big|_{0.5 * 0.3} \\
&\quad \cap \{ (d_1, 0.2), (d_3, 0.3), (d_6, 0.6), (d_8, 0.3), (d_2, 0.3), (d_4, 0.3), (d_5, 0.2), \\
&\quad \quad \quad (d_7, 0.4) \} \Big|_{0.5 * 0.3} \\
&\quad \cap \{ (d_1, 0.8), (d_4, 0.4), (d_2, 0.4), (d_5, 0.5), (d_8, 0.8), (d_7, 0.3), (d_3, 0.2), \\
&\quad \quad \quad (d_6, 0.4) \} \Big|_{0.5 * 0.4} = \\
&= \{ (d_2, 1.0), (d_3, 0.8), (d_4, 1.0), (d_5, 0.9), (d_7, 0.9), (d_8, 0.9) \}.
\end{aligned}$$

7. GEOMETRYCZNY MODEL ROZPROSZONEGO SYSTEMU WYSZUKIWANIA INFORMACJI

Pierwszą geometryczną interpretację hierarchicznych systemów wyszukiwania informacji podał Dąbrowski w pracy [28], natomiast rozszerzenie tego modelu uwzględniające dodatkowo wagi deskryptorów podano w pracy [61]. W rozdziale tym podamy geometryczną interpretację rozproszonych systemów wyszukiwania informacji.

Każdy opis dokumentu i każdą kwerendę można przekształcić na punkt pewnej przestrzeni deskryptorowej. Punkt tej przestrzeni odpowiadający kwerendzie wyznacza podzbiór dokumentów, których opisy nie mają deskryptorów ogólniejszych od żadnego z deskryptorów z kwerendy. Odpowiedź na tę kwerendę będzie więc podzbiorem wyznaczonego zbioru dokumentów.

7.1. Oznaczenia i definicje

Rozważmy globalny zbiór deskryptorów T_0 , skończony, uporządkowany częściowo przez relację R [94]. Niech $T^* \subset T_0$ będzie zbiorem wszystkich deskryptorów parami nieuporządkowanymi w sensie relacji R , tj. dla $t' \neq t''$

$$\{t', t''\} \subset T^* \iff \{t', t''\} \subset T_0 \times T_0 \setminus R.$$

Ponieważ T_0 jest zbiorem skończonym, skończony jest także zbiór T^* . Oznaczając przez s moc zbioru T^* , na mocy twierdzenia Dillwortha [133] możemy utworzyć s ciągów

$$T_i = \{t_{i1}, t_{i2}, \dots, t_{im_i}\}, \quad i=1, 2, \dots, s \quad (29)$$

spełniających warunki:

$$(i) \quad t_{im_i} R t_{im_i-1} R \dots R t_{i1}, \quad i=1, 2, \dots, s, \quad (30)$$

$$(ii) \quad \forall t \in T_0 \quad \exists 1 \leq i \leq s \quad \exists 1 \leq k \leq m_i \quad t = t_{ik}, \quad (31)$$

$$(iii) \quad T_i \cap T_j = \emptyset \quad \text{dla} \quad i \neq j \wedge 1 \leq i, j \leq s. \quad (32)$$

W każdym zbiorze T_i relacja R (a właściwie jej obcięcie) indukuje porządek liniowy. Deskryptor $t \in T_0$ oznaczamy będziemy przez t_{ik} , gdzie indeks i wskazuje, do którego ciągu T_i deskryptor t został przyporządkowany, a indeks k miejsce tego deskryptora w ciągu T_i . Do każdego T_i dołączony jest pseudodeskryptor $*$ taki, że $* \in T_0$ spełniający $tR*$ dla dowolnego $t \in T_0$.

Z warunku (3) wynika, że w każdym opisie dokumentu liczba deskryptorów jest mniejsza lub równa s . Na podstawie warunku (3) możemy również twierdzić, że poszczególne deskryptory w opisie dokumentu należą do różnych zbiorów T_i , przy ustalonym opisie $t(d)$

$$\begin{aligned} \forall t' t'' \{ (t', w'), (t'', w'') \} \subset t(d) &\Rightarrow \\ &\Rightarrow \exists 1 \leq i \leq s \exists 1 \leq k \leq s (i \neq k \wedge t' \in T_i \wedge t'' \in T_i) \end{aligned} \quad (33)$$

(Podobne wnioski słuszne są, oczywiście, i dla kwerend).

Niech Φ oznacza zbiór opisów dokumentów, Q zbiór kwerend. Dla $t(d) \in \Phi$ określimy zbiór

$$I(d) = \{ i : 1 \leq i \leq s \wedge \exists (t, w) \in t(d) t \in T_i \}.$$

Ponieważ dla każdego $i \in \{1, 2, \dots, s\}$ istnieje co najwyżej jeden deskryptor z wagą $(t, w) \in t(d) \wedge t \in T_i$, dla $i \in I(d)$, można jednoznacznie ustalić deskryptor $t(i) : (t(i), w) \in t(d) \wedge t(i) \in T_i$. Podobnie określamy zbiór $I(q)$ oraz element kwerendy $r(i)$ (aby zwiększyć czytelność zapisu, elementy kwerendy q będziemy oznaczać przez r), który wchodzi do ciągu T_i , $i \in I(q)$.

Przekształcenie opisów dokumentów w ciągi s -elementowe par (deskryptor, waga) zdefiniujemy następująco:

$$a_i(d) = \begin{cases} (t(i), w) & i \in I(d) \\ (*, 0) & i \notin I(d) \end{cases} \quad (34)$$

gdzie przez $a_i(d)$ rozumiemy i -ty element ciągu, w którym w jest wagą deskryptora $t(i)$ występującego w opisie $t(d)$ dokumentu d . Ciągi te nazywać będziemy współrzędnymi deskryptorowymi opisów dokumentów w przestrzeni dokumentów.

Podobnie przekształcamy opisy kwerend w ciągi s -elementowe par (deskryptor, waga)

$$b_i(q) = \begin{cases} (r(i), w) & i \in I(q) \\ (*, 0) & i \notin I(q) \end{cases} \quad (35)$$

gdzie $b_i(q)$ oznacza i -ty element ciągu, w którym w jest wagą deskryptora $r(i)$ występującego w kwerendzie q . Ciągi te nazywać będziemy współrzędnymi deskryptorowymi kwerend w przestrzeni dokumentów.

Dla elementu kwerendy r określimy ponadto zbiór

$$J(r) = \{j: (\exists k \in \{1, 2, \dots, m\}) (rRt_{jk})\}.$$

Na mocy warunku (31) $J(r) \neq \emptyset$. Dla $j \in J(r)$ oznaczmy przez $k(r, j)$ największe k , przy którym zachodzi jeszcze rRt_{jk} . Dla danej kwerendy q określamy s -elementowy ciąg deskryptorów postaci:

$$h_j(q) = \begin{cases} r(i) & i=j \wedge i \in I(q) \\ t(j)_1 & (\exists i \in I(q)) i \neq j \wedge j \in J(r(i)) \wedge l=k(r, j) \\ t(j)_{m_j} & \text{w pozostałych przypadkach} \end{cases} \quad (36)$$

gdzie $h_j(q)$ jest j -tym elementem ciągu $h(q)$, $t(j)_1$ oznacza 1-ty element w zbiorze deskryptorów T_j .

Ciąg $h(q)$ odpowiadający dowolnej kwerendzie q wyznacza podział tej przestrzeni na dwie podprzestrzenie D_q^1 oraz D_q^2 takie, że współrzędne punktów podprzestrzeni D_q^1 nie są deskryptorami ogólniejszymi niż deskryptory będące współrzędnymi punktu $h(q)$; natomiast co najmniej jedna współrzędna dowolnego punktu przestrzeni D_q^2 jest deskryptorem ogólniejszym niż pewien deskryptor będący współrzędną punktu $h(q)$.

Zgodnie z definicją relacji zawierania $\succ$ (6), do zbioru dokumentów będących odpowiedzią na kwerendę q , reprezentowaną przez punkt $h(q)$ deskryptorowej przestrzeni D , nie należą te dokumenty, którym odpowiadają punkty podprzestrzeni D_q^2 . Odpowiedź na tę kwerendę jest zawarta w podprzestrzeni D_q^1 .

7.2. Konstrukcja globalnej przestrzeni dokumentów

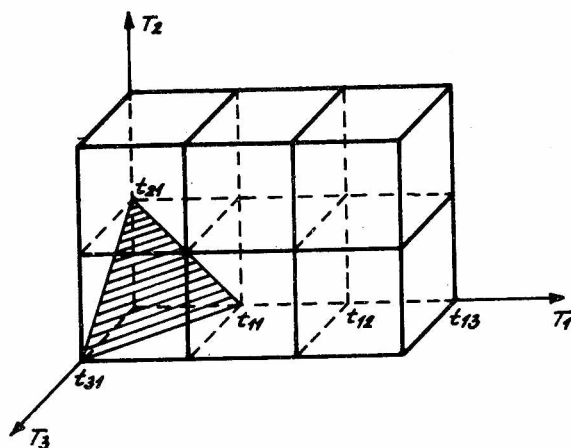
Globalna przestrzeń opisów dokumentów jest przestrzenią s -wymiarową. Pseudodeskryptor $*$ (deskryptor zerowy) jest początkiem układu współrzędnych. Zbiory T_i wyznaczają poszczególne osie. Na osiach T_i leżą deskryptory ze zbioru T_i . Deskryptory szczegółowe leżą bliżej początku układu współrzędnych, ogólniejsze dalej. Odległość pomiędzy deskryptorami na poszczególnych osiach układu wynosi 1. Deskryptory na osiach T_i ($i=1, 2, \dots, s$) wyznaczają przestrzeń deskryptorową składającą się z $m = \prod_{i=1}^s m_i$ ($m_i = \text{card}(T_i)$) jednostkowych kostek s -wymiarowych.

Opisy dokumentów traktować będziemy jako punkty, leżące wewnątrz kostek. Wszystkie odcinki $(*, t_{11}]$, $(t_{11}, t_{12}]$, $(t_{12}, t_{13}]$, ... są długości 1. Każdy odcinek utożsamiamy z jego prawym końcem. Waga deskryptora t_{ij} w opisie dokumentu określa położenie punktu względem prawego końca odcinka $(t_{ij-1}, t_{ij}]$. Deskryptory należące do opisu dokumentu (punktu w

przestrzeni dokumentów) określają kostkę, w której znajduje się dokument, natomiast wagi opisują położenie dokumentu w danej kostce. Ponieważ w każdym globalnym opisie dokumentu suma wag jest równa 1, dokumenty będą leżeć na przekątnych kostek.

Przykład 14

Niech $T_1 = \{t_{11}, t_{12}, t_{13}\}$, $T_2 = \{t_{21}, t_{22}\}$, $T_3 = \{t_{31}\}$ będą zbiorami deskryptorów. Przestrzeń rozpiętą na osiach T_1, T_2, T_3 przedstawiono na rys. 10.



Rys. 10. Przestrzeń dokumentów
Fig. 10. Document space

Zakreskowana przekątna kostki jest miejscem geometrycznym punktów, które w swoich opisach zawierają deskryptory t_{31} , t_{21} , t_{11} .

7.3. Własności deskryptorowej przestrzeni dokumentów

Każdej kostce można przyporząkować s-elementowy ciąg, który opisuje wierzchołek kostki najdalej położony względem początku układu współrzędnych. Ciąg ten nazywać będziemy opisem kostki. Otoczeniem dowolnego punktu d nazywać będziemy zbiór $X(d, c)$ postaci:

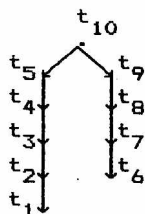
$$X(d, c) = \{d' : d' \in D \wedge |a_i(d) - a_i(d')| \leq a_i(1-c)\},$$

gdzie $a_i(d), a_i(d')$ są współrzędnymi deskryptorowymi punktów d, d' .

Dla punktu o współrzędnych $b_i(q)$ wyznaczamy wierzchołek kostki $h(q)$ wyznaczający podprzestrzeń D_q^1 kostek, których opisy składają się z deskryptorów ściślejszych od opisu kostki $h(q)$. Odpowiedzią na kwerendę $b(q)$ jest otoczenie $X(b(q), c)$ punktu $b(q)$ w podprzestrzeni D_q^1 .

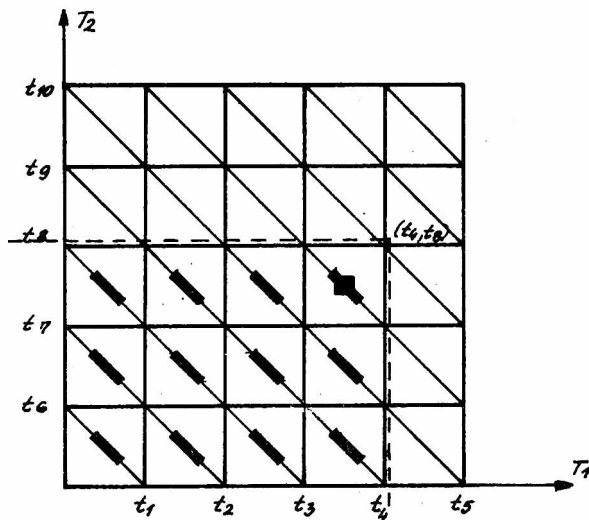
Przykład 15

Niech w rozproszonym systemie S zbiory deskryptorów T_i , ($i=1,2$) zawierają elementy: $T_1=\{t_1, t_2, t_3, t_4, t_5\}$, $T_2=\{t_6, t_7, t_8, t_9, t_{10}\}$ oraz graf relacji uogólnienia R pomiędzy deskryptorami ma postać jak na rys. 11.



Rys. 11. Graf opisujący relację uogólnienia R
Fig. 11. Graph describing relation generalization R

Przestrzeń dokumentów rozpiętą na osiach T_1 i T_2 przedstawiono na rys. 12.



Rys. 12. Przestrzeń dokumentów dla tezauryśa T
Fig. 12. Document space for thesaurus T

Niech $q = \{(t_4, 0.5), (t_8, 0.5)\}$ będzie kwerendą. Na rysunku 12 kwadracikiem ■ zaznaczono położenie punktu kwerendy w przestrzeni dokumentów. Wierzchołek kostki (t_4, t_8) wyznacza podprzestrzeń kostek, do której należą dokumenty zawierające w swoich opisach deskryptory ściślejsze od opisu kostki (t_4, t_8) . Otoczenia kwerendy w poszczególnych kostkach zaznaczono pogrubionymi odcinkami. Przyjmujemy, że dokładność odpowiedzi wynosi $c=0.5$.

7.4. Lokalne podprzestrzenie opisów dokumentów

Z każdego zbioru T_i wybieramy deskryptory należące do podsystemu S_j , otrzymując w ten sposób zbiory $T_{ij} \subset T_i$. Wszystkie własności zbiorów T_i zostaną zachowane. W każdym podzbiorze T_{ij} relacja R_j indukuje porządek liniowy. Do każdego T_{ij} dołączony jest również pseudodeskryptor $*$. Liczba s_j zbiorów T_{ij} jest mniejsza lub równa mocy zbioru T^* .

Lokalna podprzestrzeń opisów dokumentów rozproszonego systemu wyszukiwania informacji jest przestrzenią s_j -wymiarową. Pseudodeskryptor $*$ jest początkiem układu współrzędnych. Zbiory T_{ij} wyznaczają poszczególne osie. Deskryptory szczegółowe leżą bliżej początku układu współrzędnych.

Przykład 16

Założmy, że w podsystemie S_1 tezaurus T_1 zawiera deskryptory $\{t_1, t_2, t_4, t_5, t_6, t_7, t_9, t_{10}\}$, a w podsystemie S_2 tezaurus T_2 zawiera deskryptory $\{t_2, t_3, t_4, t_7, t_8\}$. Grafy relacji uogólnienia R_j w podsystemach S_j ($j=1,2$) mają postać tak jak na rys. 13.



Rys. 13. Graf relacji uogólnienia R_1, R_2 w podsystemach S_1, S_2
 Fig. 13. Graph of relation generalization R_1, R_2 in subsystems S_1, S_2

Tak więc zbiory T_{ij} zawierają następujące elementy:

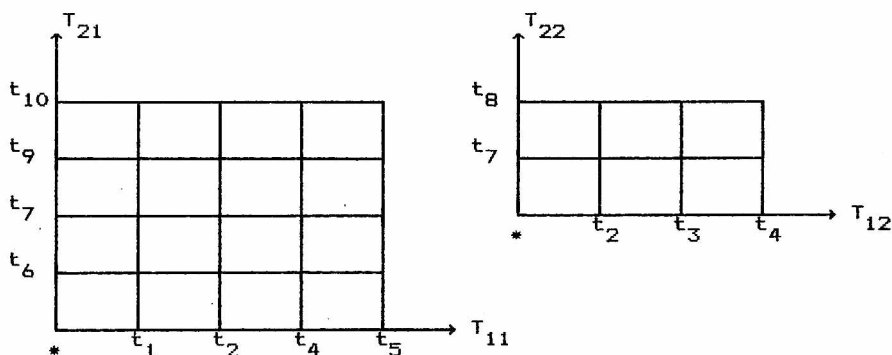
$$T_{11} = \{t_1, t_2, t_4, t_5\}$$

$$T_{12} = \{t_2, t_3, t_4\}$$

$$T_{21} = \{t_6, t_7, t_9, t_{10}\}$$

$$T_{22} = \{t_7, t_8\}$$

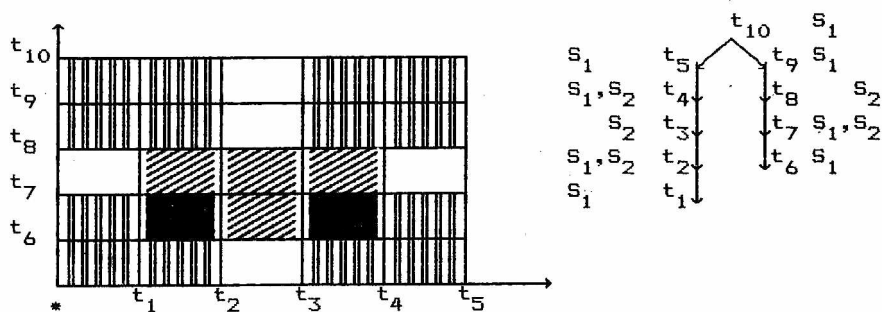
Przestrzenie dokumentów dla podsystemów S_1 i S_2 rozpięte na osiach T_{11} , T_{21} i T_{12} , T_{22} mają postać przedstawioną na rys. 14.



Rys. 14. Przestrzenie dokumentów w podsystemach S_1 , S_2

Fig. 14. Document spaces in subsystems S_1 , S_2

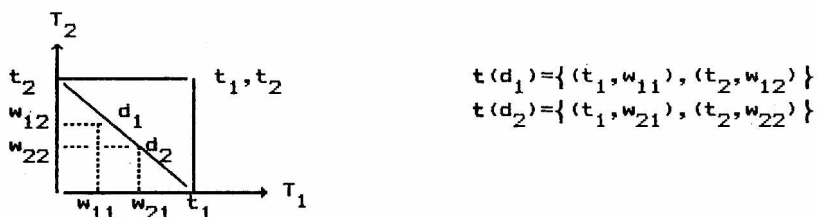
Na rys. 15 kreskami pionowymi zaznaczono kostki należące do podprzestrzeni systemu S_1 , kreskami ukośnymi zaznaczono kostki należące do podprzestrzeni systemu S_2 , natomiast ciemnymi polami oznaczono kostki należące jednocześnie do podprzestrzeni systemów S_1 i S_2 .



Rys. 15. Rozproszona przestrzeń dokumentów

Fig. 15. Distributed document space

Rzutek dowolnego punktu (dokumentu) w deskryptorowej przestrzeni dokumentów na jedną z osi współrzędnych T_i nazywać będziemy parę (dokument, waga) (rys. 16).



Rys. 16. Rzuty punktów w przestrzeni dokumentów
Fig. 16. Projections of points in document space

Zbiory tych par są listami inwersyjnymi dla deskryptorów leżących na osiach współrzędnych T_i . Taka interpretacja rzutów dokumentów jest zgodna z definicją list inwersyjnych podaną w rozdziale 4. Rzuty są pamiętane w tych podsystemach, w których występują dane deskryptory. Dokumenty zawierające w swoich opisach deskryptory t' , t'' będą należały do kostek w przestrzeni globalnego systemu, które w swoich opisach zawierają również te deskryptory. Gdy oba deskryptory należą do różnych podsystemów, dokument o takim opisie nie istnieje w tych podsystemach. Wtedy w podsystemach pamięta się tylko rzuty dokumentów na osie współrzędnych, na których występują te deskryptory.

Na rys. 15. można zauważyć, że przestrzeń dokumentów dla globalnego systemu zawiera "dziury", dokumenty z podsystemów mogą nie wypełniać całej przestrzeni dla globalnego systemu. Położenia dokumentów w "pustych" kostkach odtwarza się na podstawie operacji iloczynu mnogościowego rzutów z lokalnych podsystemów. Odpowiada to operacji przecięcia odpowiednich list inwersyjnych, tzn. globalna odpowiedź tworzona jest na podstawie operacji na odpowiedziach z lokalnych podsystemów.

8. MODYFIKACJA PRZESTRZENI DOKUMENTÓW

8.1. Metody modyfikacji przestrzeni dokumentów

Efektywność wyszukiwania informacji osiągnięta obecnie nie zawsze spełnia oczekiwania większości potencjalnych użytkowników. Najbardziej widoczną poprawę efektywności wyszukiwania można uzyskać za pomocą adaptacyjnych metod współdziałania użytkownika z systemem. Pozwalają one na reorganizację danych na podstawie informacji otrzymanej od użytkownika.

Jedną z technik polepszenia efektywności automatycznego wyszukiwania informacji jest modyfikacja kwerend oparta na subiektywnych ocenach relewancji dokumentów wcześniej odnalezionych i dostarczonych użytkownikom. Różne badania modyfikacji kwerend i ich wyniki szczegółowo przedstawiono w pracach [86], [88].

Metody zwiększania efektywności wyszukiwania przez modyfikację kwerend mają pewne wady. Główną z nich jest to, że modyfikacje kwerend nie mają stałego charakteru, nawet jeśli mogą one znacznie poprawić obsługę konkretnego zapytania użytkownika [86], to jednak nie mogą ulepszyć samego systemu. Inne metody wprowadzające pożyteczne stałe zmiany w systemie mogą przynieść stałe polepszanie efektywności wyszukiwania. Jedyną stałą strukturą w systemie jest przestrzeń wektorów opisów dokumentów, która powinna być przedmiotem modyfikacji.

Zmiany w przestrzeni opisów dokumentów mogą mieć wielorakie przeznaczenie:

(i) Kwerendom jednolitej pod względem specjalizacji zawodowej grupy użytkowników odpowiadają podobne zbiory dokumentów. Wykorzystanie tego faktu może uprościć wyszukiwanie dokumentów relewantnych dla podobnych kwerend stawianych w przyszłości.

(ii) Mimo standaryzacji, słownictwo wielu dziedzin naukowych nie pozostaje stałe, zmienia się wraz z rozwojem nowych dziedzin nauki, napływem personelu i innych czynników. W rezultacie po kilku latach może się okazać, że opisy dokumentów straciły na swej aktualności. Metoda modyfikacji polegająca na wprowadzaniu nowych pojęć do opisów dokumentów na podstawie zgłaszanych kwerend może zapewnić stałą aktualizację

słownictwa występującego w tych opisach.

(iii) Wagi deskryptorów w opisach są ustalane najczęściej zgodnie z częstością występowania pojęć w streszczeniach danych dokumentów. Ustalanie wag sprowadza się więc do prostego procesu przeliczania. Jednakże założenie, że najważniejsze są pojęcia najczęściej powtarzające się w streszczeniu nie zawsze jest właściwe. Aktualizacja wag wielu pojęć dokumentu na podstawie subiektywnej opinii użytkownika pozwala utworzyć opisy dokumentów lepiej odzwierciedlające rzeczywistość ich treść.

Jest wiele metod usprawniania wyszukiwania za pomocą modyfikacji opisów dokumentów. Poniższe metody (37)–(41) zaproponowali Ide i Brauen [88] przy badaniach zbiorów Cranfield.

Po otrzymaniu odpowiedzi na swoją kwerendę użytkownik ocenia, które z uzyskanych dokumentów są relewantne, a które nie. Oceny użytkownika są przesyłane do systemu na zasadzie sprzężenia zwrotnego. Zgodnie z tymi wskazówkami użytkownika modyfikowane są opisy dokumentów, co można przeprowadzić według następujących metod:

(i) Jeśli użytkownik oceni dokument należący do odpowiedzi jako relewantny względem jego kwerendy q , to opis dokumentu $t(d)$ zostanie przekształcony za pomocą wzoru

$$t(d) := (1-\lambda)t(d) + \lambda q \quad (37)$$

Gdy dokument oceniony jest jako nierелеwantny, wówczas jego opis jest przekształcony według wzoru

$$t(d) := (1+\lambda)t(d) - \lambda q \quad (38)$$

(ii) Do modyfikacji opisów relewantnych i nierелеwantnych dokumentów stosuje się odpowiednie wzory

$$t(d) := (1-\lambda)t(d) + \lambda q \quad (39)$$

oraz

$$t(d) := (1-\lambda)t(d) - \lambda q \quad (40)$$

(iii) Modyfikuje się tylko opisy relewantnych dokumentów według wzoru

$$t(d) := (1-\lambda)t(d) + \lambda q \quad (41)$$

Opisy dokumentów nierелеwantnych pozostają bez zmian.

Parametr λ występujący we wszystkich wzorach (37)–(41) charakteryzuje szybkość zmian wag w opisach dokumentów w procesie modyfikacji. Przeprowadzone badania przez Brauena [88] wykazały, że najlepsze rezultaty można otrzymać dla $\lambda \leq 0.2$. Wyższe wartości λ prowadzą do błędnych modyfikacji opisów dokumentów.

Wzory (37)–(41) modyfikujące opisy dokumentów nie uwzględniają

relacji uogólnienia. Podano je w celu wyjaśnienia postaci wzorów (42)–(44) modyfikujących wagi opisów i uwzględniających relację uogólnienia.

Niech $t(d) = \{(t_1, w_1), (t_2, w_2), \dots, (t_n, w_n)\}$ będzie opisem dokumentu, a $q = \{(r_1, v_1), (r_2, v_2), \dots, (r_m, v_m)\}$ kwerendą oraz zbiór $N_i = \{k: (r_i = t_k \vee r_i Rt_k) \wedge 1 \leq k \leq n\}$ dla $1 \leq i \leq m$.

Po uwzględnieniu relacji uogólniania wzór (37) przyjmuje postać

$$\forall_{1 \leq k \leq n} w_k := \begin{cases} (1-\lambda)w_k + \frac{w_k}{\sum_{j \in N_i} w_j} \lambda v_i & \text{jeśli } r_i = t_k \vee r_i Rt_k \\ (1-\lambda)w_k & \text{w p.p.} \end{cases} \quad (42)$$

wzór (38) przyjmuje postać

$$\forall_{1 \leq k \leq n} w_k := \begin{cases} \left| (1+\lambda)w_k - \frac{w_k}{\sum_{j \in N_i} w_j} \lambda v_i \right| & \text{jeśli } r_i = t_k \vee r_i Rt_k \\ (1+\lambda)w_k & \text{w p.p.} \end{cases} \quad (43)$$

oraz ze wzoru (40) uzyskamy

$$\forall_{1 \leq k \leq n} w_k := \begin{cases} \left| (1-\lambda)w_k - \frac{w_k}{\sum_{j \in N_i} w_j} \lambda v_i \right| & \text{jeśli } r_i = t_k \vee r_i Rt_k \\ (1-\lambda)w_k & \text{w p.p.} \end{cases} \quad (44)$$

Aby uniknąć ujemnych wartości wag, które niekiedy mogłyby się pojawić we wzorach (43) i (44), obliczamy wartości bezwzględne różnic. Normalizacji wag w opisach dokonujemy po modyfikacjach wag według wzoru (44) lub jeśli we wzorze (43) przynajmniej jedna z różnic, dla dowolnego j , miała wartość ujemną, tj.

$$\forall_{1 \leq k \leq n} w'_k := \frac{w_k^*}{\sum_{i=1}^n w'_i + 1 - \sum_{i=1}^n w_i},$$

gdzie w oznacza wagę deskryptora w opisie przed modyfikacją, a w' oznacza wagę po modyfikacji.

Modyfikacja opisów dokumentu przeprowadzona według wzorów (42)–(44) pozwala na zmianę wag i usuwanie deskryptorów, których wagi zmalały do zera. Powyższe metody nie umożliwiają dołączanie nowych deskryptorów do opisów dokumentów ze względu na konieczność spełnienia warunku $q \succ t(d)$. Wadę powyższych metod usuwa się poprzez dodatkowe modyfikacje opisów dokumentów. Przy przesłaniu do systemu oceny o relewantności dokumentu użytkownik może podać nowy deskryptor z wagą, który według niego powinien wystąpić w opisie tego dokumentu. Dołączony deskryptor nie może

być ani ogólniejszy, ani szczegółowszy od wszystkich pozostałych deskryptorów występujących w opisie dokumentu. W tym przypadku modyfikacja wektora opisu dokumentu przebiega według wzoru

$$w_i := \lambda w' \quad (45)$$

gdzie w' jest wagą nowego deskryptora.

Po dołączeniu nowego deskryptora z wagą normalizuje się opis dokumentu, tj.

$$\forall \quad w_k := \frac{w_k}{1 + w_i}, \quad 1 \leq k \leq n$$

gdzie i jest wskaźnikiem dołączanego deskryptora.

Użytkownicy wymagający mniej dokładnych odpowiedzi powinni mieć mniejszy wpływ na modyfikację opisów dokumentów niż użytkownicy wymagający odpowiedzi bardzo dokładnych. W odpowiedziach o małej dokładności znajdują się dokumenty, których stopień podobieństwa do kwerend jest mały. Zwiększa się wtedy możliwość nieprawidłowej klasyfikacji wśród różnych orzekających o relewantności dokumentu. Dlatego też we wzorach (42)–(44) za parametr λ można przyjąć $\lambda = 0.1(c+1)$. Użytkownicy wymagający dużych dokładności c odpowiedzi mają wówczas większy wpływ na modyfikację opisów dokumentów.

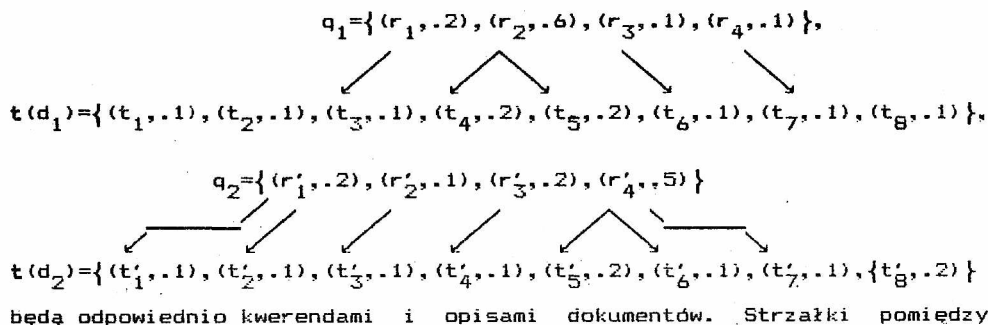
Bardzo prosta postać wzorów (42)–(44) pozwala na łatwe stosowanie w praktyce podanych wyżej metod modyfikacji opisów dokumentów.

Modyfikacja opisów dokumentów relewantnych (nierелеwantnych) ma na celu zwiększenie (zmniejszenie) stopnia zawierania się kwerendy q w opisie dokumentu $t(d)$.

W przykładzie 16 pokażemy jak szybko wzrasta stopień zawierania się kwerendy w opisach dokumentów.

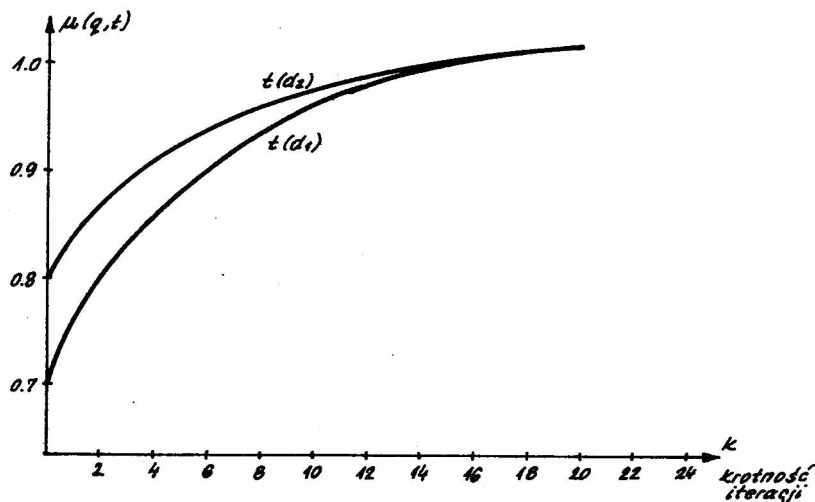
Przykład 16

Niech



deskryptorami kwerend i opisów wskazują relację uogólnienia R.

Zakładamy, że dokumenty d_1, d_2 oceniane są jako relewantne względem kwerend q_1, q_2 . Modyfikację przeprowadzono według wzoru (42), za wartość parametru λ przyjęto 0.16. Na rys. 17 przedstawiono wykres zależności stopnia zawierania μ od liczby dokonanych modyfikacji opisu dokumentu.



Rys. 17. Zależność stopnia zawierania μ od krotności iteracji
Fig. 17. Dependence of degree μ and number of iterations

Po 20 iteracjach opisy $t(d_1), t(d_2)$ przyjmują postać:

$$t(d_1) = \{(t_1, .003), (t_2, .003), (t_3, .197), (t_4, .297), (t_5, .297), (t_6, .1)\}$$

$$t(d_2) = \{(t'_1, .1), (t'_2, .1), (t'_3, .1), (t'_4, .197), (t'_5, .248), (t'_6, .124), (t'_7, .124), (t'_8, .006)\}.$$

W wyniku modyfikacji opisów dokumenty ocenione jako relewantne zostaną przysunięte do kwerendy, a nirelewantne odsunięte od kwerendy.

Przykład 17

Założmy, że dana jest kwerenda $q = \{(t_1, 0.2), (t_2, 0.8)\}$ oraz opis dokumentu $t'(d) = \{(t_1, 0.7), (t_2, 0.3)\}$. Wówczas stopień zawierania się kwerendy q w opisie dokumentu wynosi $\mu(q, t(d)) = 0.5$.

Jeśli dokument d zostanie oceniony jako relewantny względem kwerendy, to w wyniku modyfikacji według wzoru (42) dla parametru $\lambda = 0.2$

opis dokumentu przyjmie postać:

$$t'(d) = \{(t_1, 0.7 \cdot 0.8 + 0.2 \cdot 0.2), (t_2, 0.3 \cdot 0.8 + 0.8 \cdot 0.2)\} = \{(t_1, 0.6), (t_2, 0.4)\}$$

oraz $\mu(q, t') = 0.6$.

Gdy dokument d zostanie oceniony jako nierelevantny, wówczas w wyniku modyfikacji według wzoru (43) dla $\lambda = 0.2$ otrzymamy opis

$$t''(d) = \{(t_1, 0.7 \cdot 1.2 - 0.2 \cdot 0.2), (t_2, 0.3 \cdot 1.2 - 0.8 \cdot 0.2)\} = \{(t_1, 0.8), (t_2, 0.2)\}$$

i $\mu(q, t'') = 0.4$.

8.2. Symulacja procesu wyszukiwania informacji w rozproszonym systemie

W rozdziale tym pokażemy, że modyfikowanie przestrzeni opisów dokumentów w lokalnych podsystemach przynosi korzyści użytkownikom w postaci zwiększenia efektywności systemu. Analityczne badanie zmian efektywności jest bardzo trudne do wykonania, dlatego też będziemy opierać się na badaniach empirycznych.

Aby zbadać zachowanie się rozproszonego systemu w czasie modyfikacji lokalnych opisów dokumentów, oceny użytkowników zastąpiono pewnym mechanizmem programowym, który zasymulował praktyczne zachowanie się systemu. Każdy dokument znajdujący się w przestrzeni opisów dokumentów ma swój obraz będący opisem idealnym (eksperckim). Opisy dokumentów modyfikuje się po to, aby przybliżyć je do opisów idealnych.

Dany dokument d o opisie $t(d)$ należący do odpowiedzi na kwerendę q jest porównywany ze swoim idealnym obrazem $t_e(d)$. Jeśli opis $t(d)$ jest mniej podobny (mniejszy stopień zawierania μ) do kwerendy niż opis idealny $t_e(d)$, to podejmuje się decyzję o relewantności dokumentu w celu przesunięcia w przestrzeni jego opisu bliżej kwerendy Q , a tym samym bliżej opisu idealnego.

Dokument d o opisie $t(d)$ będziemy uważać za relewantny względem kwerendy q , jeśli

$$(i) \mu(q, t_e(d)) \geq \mu(q, t(d))$$

lub

$$(ii) \mu(q, t_e(d)) < \mu(q, t(d)) \wedge \mu(q, t_e(d)) \geq \mu(t_e(d), t(d)).$$

W pozostałych przypadkach dokument o opisie $t(d)$ będziemy uważać za nierelevantny.

Symulację pracy rozproszonego systemu przeprowadzono na

mikrokomputerze IBM PC. Na bazie dwóch lokalnych podsystemów wyszukiwania informacji opartych na tezaurusach z wagami zbudowano rozproszony system, w którym globalne odpowiedzi były tworzone jako wyniki operacji na częściowych odpowiedziach z lokalnych podsystemów. W podsystemie S_1 tezaurus T_1 zawierał 26 deskryptorów, w podsystemie S_2 tezaurus T_2 zawierał 24 deskryptory, globalny tezaurus $T=T_1 \cup T_2$ zawierał 41 deskryptorów.

8.3. Analiza wyników adaptacji rozproszonego systemu

W rozdziale tym zostaną podane definicje współczynników korelacji porównujących opisy dokumentów przed i po modyfikacji z opisami idealnymi dokumentów należących do odpowiedzi na tę samą kwerendę. Zadaniem tych współczynników jest pokazanie, w jakim stopniu "korelują" ze sobą odpowiedzi przed i po adaptacji systemu. Zbadamy, czy różnice pomiędzy odpowiednimi współczynnikami są przypadkowe czy też istotne.

Dla każdego zbioru: opisów dokumentów przed adaptacją, opisów po adaptacji i opisów idealnych utworzono listy inwersyjne. W ten sposób w każdym podsystemie otrzymano trzy rodzaje list inwersyjnych. Za pomocą tych list wyszukiwano odpowiedzi na 15 kwerend nie biorących udziału w procesie modyfikacji. Miarę podobieństwa ρ , porównującą odpowiedzi na tę samą kwerendę, można zdefiniować następująco:

Definicja 17

Niech zbiór X zawiera numery (adresy) dokumentów należących do odpowiedzi w deskryptorowej przestrzeni opisów idealnych, zbiór Y zawiera numery dokumentów należących do odpowiedzi w przestrzeni opisów przed modyfikacją, zbiór Z numery dokumentów po modyfikacji. Określone są funkcje

$$f_1, g_1: X \cup Y \rightarrow [0, 1],$$

$$f_2, g_2: X \cup Z \rightarrow [0, 1],$$

takie, że

$$f_1(d) = \begin{cases} \mu(q, t_e(d)) & d \in X, \\ 0 & d \notin X, \end{cases}$$

$$g_1(d) = \begin{cases} \mu(q, t(d)) & d \in Y, \\ 0 & d \notin Y, \end{cases}$$

$$f_2(d) = \begin{cases} \mu(q, t_e(d)) & d \in X, \\ 0 & d \notin X, \end{cases}$$

$$g_2(d) = \begin{cases} \mu(q, t_m(d)) & d \in Z \\ 0 & d \notin Z \end{cases}$$

Zbiory wartości f_1 i f_2 są jednakowe, ale dziedziny tych funkcji są różne. Zachowanie się różnic stopni zawierania $\mu(q, t)$ dla różnych kwerend w zbiorach X i Y oraz X i Z będziemy badać przez porównywanie współczynników korelacji określonych na zbiorach X , Y i Z .

$$\rho(X, Y) = 1 - \frac{\sum_{d \in X \cup Y} |f_1(d) - g_1(d)|}{\sum_{d \in X} f_1(d) + \sum_{d \in Y} g_1(d)},$$

$$\rho(X, Z) = 1 - \frac{\sum_{d \in X \cup Z} |f_2(d) - g_2(d)|}{\sum_{d \in X} f_2(d) + \sum_{d \in Z} g_2(d)}.$$

Jeśli $X \cap Y = \emptyset$, to $\rho(X, Y) = 0$ oraz jeśli $X = Y$ i miary $\mu(q, t)$, $\mu(q, t_e)$ są równe, to $\rho(X, Y) = 1$.

Modyfikację opisów dokumentów relewantnych przeprowadzono według wzoru (42). Parametr λ wynosił 0.18, a odpowiedzi były wyszukiwane z dokładnością $c=0.2$. Lokalne podsystemy S_1 i S_2 zawierały po 100 opisów dokumentów oraz po 100 odpowiadających im idealnych opisów dokumentów.

Badania nad adaptacją rozproszonego systemu prowadzono w kierunku otrzymania odpowiedzi na dwa pytania:

1. Czy modyfikacje opisów dokumentów w lokalnych podsystemach są również korzystne dla globalnych użytkowników?
2. Czy orzeczenia o relewancji dokumentów przez globalnych użytkowników i związane z tą oceną modyfikacje globalnej przestrzeni mają korzystny wpływ na podsystemy i ich lokalnych użytkowników?

Wpływ modyfikacji opisów dokumentów w scentralizowanym oraz w lokalnym podsystemie na efektywność systemu szczegółowo przedstawiono w pracy [61]. Przy pomocy badań przeprowadzonych na próbie losowej stwierdzono, że w zbiorowości generalnej współczynniki korelacji porównujące odpowiedzi po modyfikacji opisów są większe niż przed modyfikacją oraz że większej liczbie modyfikacji opisów dokumentów odpowiada większy przyrost współczynników korelacji odpowiedzi. Przyrosty współczynników zależą również od opisów dokumentów. Jeśli pomiędzy kwerendą a opisami dokumentów występują duże różnice wag i deskryptorów (mały stopień zawierania się kwerendy w opisie dokumentu), to przyrosty

współczynników korelacji odpowiedzi są większe niż gdy pomiędzy opisami i kwerendami występują małe różnice. W pierwszym okresie modyfikacji opisów dokumentów przyrosty współczynników korelacji odpowiedzi są większe niż przed reorganizacją list inwersyjnych. W czasie modyfikacji opisy dokumentów "upodabniają" się do kwerend.

8.3.1. Modyfikacja lokalnych podprzestrzeni dokumentów a efektywność rozproszonego systemu

W celu znalezienia odpowiedzi na pierwsze pytanie modyfikowano lokalne podsystemy niezależnie od siebie. W obu podsystemach do adaptacji podsystemów użyto po 35 lokalnych kwerend. Aby określić wpływ lokalnych modyfikacji na efektywność rozproszonego systemu, porównano odpowiedzi na 15 globalnych kwerend przed i po modyfikacji podsystemów. Żadna z tych kwerend nie brała udziału w modyfikowaniu podsystemów. Z obliczonych stopni zawierania $\mu(q, t(d))$ dla par (kwerenda, dokument) przed modyfikacją oraz po modyfikacji $\mu(q, t_g(d))$ otrzymujemy twierdzącą odpowiedź na pierwsze pytanie, że w wyniku modyfikacji lokalnych podsystemów globalne opisy dokumentów upodabniają się do globalnych opisów idealnych.

W tabeli 1 Dodatku podano współczynnik podobieństwa ρ dla odpowiedzi na 15 globalnych kwerend. Współczynnik $\rho(X, Y)$ jest współczynnikiem podobieństwa przestrzeni opisów przed adaptacją do przestrzeni opisów idealnych, $\rho(X, Z)$ jest współczynnikiem podobieństwa przestrzeni po adaptacji do przestrzeni opisów idealnych.

Do obliczania średnich wartości współczynników ρ nie stosowano średnich arytmetycznych, ale średnie ważone. Ponieważ odpowiedzi składają się z różnej liczby dokumentów, więc odpowiedzi z większą liczbą dokumentów mają większy wpływ na średnie wartości współczynników. Średnie ważne współczynników z tabeli 1 wynoszą $\bar{\rho}(X, Y) = 0.89$, $\bar{\rho}(X, Z) = 0.953$.

Przy pomocy wnioskowania statystycznego ustalono, czy wielkości współczynników z tabeli 1 przed adaptacją i po adaptacji systemu różnią się między sobą "w sposób istotny", czy występujące między nimi różnice są przypadkowe. Różnicę współczynników można uważać za istotną, jeśli obie próby pochodzą ze zbiorowości generalnych o różnych parametrach δ_1 , δ_2 (gdzie δ_1 , δ_2 oznaczają średnie wartości współczynników odpowiednio przed i po adaptacji). W celu sprawdzenia istotności różnic sformułowano

hipotezę zerową, wyrażającą przypuszczenie, że dwie próby losowe różnią się między sobą nie w sposób "istotny", lecz tylko przypadkowy. Wyraża się to następująco $H_0: \delta_1 - \delta_2 = 0$.

Ponieważ porównywane współczynniki podobieństwa są obliczone dla odpowiedzi na te same kwerendy, nie brano pod uwagę wartości współczynników, lecz ich różnice. Hipotezę H_0 , czy średnia ważona różnic $d = \rho(X, Z) - \rho(X, Y)$ odbiega od zera w sposób przypadkowy czy też w sposób istotny, sprawdzono przy pomocy testu "t".

Niech $d_i = \rho_i(X, Z) - \rho_i(X, Y)$ będzie różnicą współczynników dla i -tej kwerendy. Obliczoną wartość t

$$t = \frac{\bar{d}}{\sqrt{\left(\sum_{i=1}^n d_i^2 - n\bar{d}^2 \right) / (n(n-1))}} \quad (46)$$

(gdzie $\bar{d}$ jest średnią ważoną różnic d_i) porównujemy z wartością ustaloną za pomocą tabeli statystycznej, z uwzględnieniem przyjętego prawdopodobieństwa błędu α oraz liczby stopni swobody $n-1$, stąd otrzymujemy następujące kryterium: jeśli $t < t_{\alpha, n-1}$, to H_0 należy przyjąć, w przeciwnym przypadku należy odrzucić. Ze względu na tradycyjnie przyjęte oznaczenia w statystyce, literą t oznacza się zmienną losową t-Studenta. W tym miejscu nie powinny powstawać żadne niejasności związane z oznaczaniem deskryptorów w poprzednich rozdziałach przez t .

Dla $d_i = \rho_i(X, Z) - \rho_i(X, Y)$, ($i=1, 2, \dots, 15$) obliczając t ze wzoru (46), otrzymujemy $t=3.174$. Odczytana z tabeli [110, s.112] dla testu jednostronnego i $\alpha=0.001$ oraz $n-1=14$ wartość $t_{\alpha, n-1}$ wynosi 2.9768. Ponieważ $t \geq t_{0.001, 14}$, hipotezę zerową należy odrzucić. Stąd otrzymujemy wniosek, że współczynniki podobieństwa ρ przed adaptacją i po adaptacji systemu są istotnie różne z prawdopodobieństwem błędu $\alpha=0.001=0.1\%$.

Podobne porównania przeprowadzono dla list inwersyjnych przed i po adaptacji względem idealnych list inwersyjnych. W tabeli 2 Dodatku podano współczynnik podobieństwa ρ dla list inwersyjnych w podsystemach S_1, S_2 w przypadku modyfikacji lokalnych podprzestrzeni dokumentów. Średnie ważne współczynniki z tabeli 2 wynoszą w podsystemie

$$S_1: \quad \bar{\rho}(X, Y)=0.787, \quad \bar{\rho}(X, Z)=0.823;$$

$$S_2: \quad \bar{\rho}(X, Y)=0.825, \quad \bar{\rho}(X, Z)=0.851.$$

Sprawdzając "istotność" różnic współczynników przed i po adaptacji dla list inwersyjnych, obliczono ze wzoru (46) dla $d_i = \rho_i(X, Z) - \rho_i(X, Y)$ wartość t , dla podsystemu S_1 otrzymano $t=6.197$, a dla S_2 $t=3.286$. Odczytana z tabeli [110, s.112] dla podsystemu S_1 wartość $t_{\alpha, n-1}$ dla $\alpha=0.005$ i $n-1=24$ wynosi 3.0905, natomiast dla podsystemu S_2 wartość

$t_{\alpha, n-1}$ dla $\alpha=0.005$ i $n-1=23$ wynosi 3.1040. Ponieważ w obu podsystemach $t \geq t_{\alpha, n-1}$, stąd wnioskujemy, że współczynniki ρ dla list inwersyjnych są istotnie różne z prawdopodobieństwem błędu $\alpha=0.005=0.5\%$.

8.3.2. Modyfikacja rozproszonej przestrzeni dokumentów a efektywność lokalnych podsystemów

W rozproszonej przestrzeni dokumentów do modyfikacji opisów dokumentów użyto 35 globalnych kwerend. Do zbadania wpływu modyfikacji globalnych opisów dokumentów na odpowiedzi w podsystemach porównano odpowiedzi na 15 lokalnych kwerend przed i po modyfikacji globalnej przestrzeni dokumentów. Z obliczonych miar podobieństwa przed i po modyfikacji otrzymujemy potwierdzenie przypuszczenia, że w wyniku modyfikacji globalnej przestrzeni dokumentów lokalne opisy dokumentów upodabniają się do lokalnych opisów idealnych (eksperymentalnych).

W tabeli 3 dodatku podano współczynniki podobieństwa q dla odpowiedzi na 15 lokalnych kwerend w podsystemach S_1 i S_2 . Średnie ważone współczynników z tabeli 3 wynoszą w podsystemie

$$S_1: \bar{\rho}(X, Y)=0.900; \bar{\rho}(X, Z)=0.923,$$

$$S_2: \bar{\rho}(X, Y)=0.924; \bar{\rho}(X, Z)=0.933.$$

Sprawdzając "istotność" różnic współczynników przed i po adaptacji, obliczamy ze wzoru (46) wartość t . Dla podsystemu S_1 wynosi ona 4.175, a dla podsystemu S_2 1.269. Po odczytaniu z tabeli [110, s.112] wartość $t_{\alpha, n-1}$:

(i) dla podsystemu S_1 , gdy $\alpha=0.005$ i $n-1=13$, $t_{\alpha, n-1}=3.3725$; stąd wnioskujemy, że współczynniki ρ są istotnie różne z prawdopodobieństwem błędu $\alpha=0.005=0.5\%$,

(ii) dla podsystemu S_2 : dla $\alpha=0.22$ i $n-1=14$ oraz $t_{\alpha, n-1}=1.296$ współczynniki ρ są istotnie różne z prawdopodobieństwem błędu $\alpha=0.22=22\%$.

W przypadku podsystemu S_2 duże prawdopodobieństwo błędu wynika z tego, że jedna z kwerend składała się z tylko jednego deskryptora o dużym stopniu ogólności i odpowiedź na tę kwerendę zawierała dużo dokumentów, dlatego też średnia ważona współczynników ρ_i zniekształciła wynik.

W tabeli 4 Dodatku podano współczynniki podobieństwa ρ dla list inwersyjnych przed i po adaptacji w podsystemach S_1 i S_2 . Średnie ważone współczynniki wynoszą:

w podsystemie S_1

$$\bar{q}(X,Y)=0.792,$$

$$\bar{q}(X,Z)=0.802,$$

$$t=4.057,$$

$$\text{dla } \alpha=0.005 \text{ i } n-1=24$$

$$t_{\alpha,n-1}=3.0905$$

w podsystemie S_2

$$\bar{q}(X,Y)=0.811,$$

$$\bar{q}(X,Z)=0.830,$$

$$t=3.211,$$

$$\text{dla } \alpha=0.005 \text{ i } n-1=23$$

$$t_{\alpha,n-1}=3.104$$

więc współczynniki podobieństwa są istotnie różne, z prawdopodobieństwem błędu 0.5% .

8.4. Uwagi

Niezależne modyfikacje w różnych podprzestrzeniach dokumentów mogą doprowadzić do niespełnienia warunku zgodności (15) opisów dokumentów. W trakcie eksploatacji systemu niespełnienie warunku (15) nie wpływa na jakość pracy systemu, bowiem pomiędzy reorganizacjami systemu listy inwersyjne pozostają bez zmian.

W trakcie reorganizacji rozproszonego systemu należy w pierwszym kroku doprowadzić lokalne opisy dokumentów do spełnienia warunku (15) np. przez obliczanie średniej arytmetycznej wag deskryptorów danego dokumentu występującego w różnych podsystemach i normalizację wag w poszczególnych opisach. Takie rozwiązanie zostało przyjęte w badanym podsystemie. Dopiero wtedy tworzy się listy inwersyjne dla aktualnych opisów dokumentów.

Jeśli reorganizację list inwersyjnych będziemy przeprowadzać po każdej kwerendzie, to użytkownicy będą mogli korzystać z bieżącej modyfikacji przestrzeni opisów dokumentów. Powiemy wtedy, że system jest bardzo "wrażliwy" na modyfikację opisów dokumentów należących do odpowiedzi na poszczególne kwerendy. Z drugiej strony duża "wrażliwość" systemu wiąże się z dużymi kosztami eksploatacji systemu.

Jeśli natomiast reorganizację list będziemy przeprowadzać rzadziej, to system będzie mniej "wrażliwy" na pojedyncze modyfikacje opisów dokumentów. Wtedy system uwzględnia "uśrednione" informacje od użytkowników i nie jest "wrażliwy" na pomyłki w ocenie dokumentów przez pojedynczych użytkowników oraz koszt eksploatacji takiego systemu może być kilkakrotnie mniejszy niż w systemach, w których reorganizacji dokonuje się po każdej kwerendzie.

W systemie można dokonywać wyboru częstotliwości przeprowadzania reorganizacji list inwersyjnych. W pracy [61] oszacowano koszty modyfikacji opisów dokumentów i reorganizacji list inwersyjnych w scentralizowanych i lokalnych podsystemach opartych na tezaurusach z wagami. W ogólnych kosztach adaptacji rozproszonego systemu należy uwzględnić dodatkowo znaczne koszty związane z rozproszeniem dokumentów. Koszty te oszacowano w rozdziale 10.

Postawmy sobie również pytanie, czy współczynniki charakteryzujące efektywność systemu są zależne od różnych zbiorów orzeczeń o relewancji, wiadomo bowiem, że na podjęcie decyzji o relewancji danego dokumentu mogą wpływać różne czynniki, np. tematyka, stopień trudności dokumentów, dostępny czas, rozmiar zbioru dokumentów, doświadczenie i wykształcenie orzekającego, sposób orzekania o relewancji itp.

Jeżeli miarą efektywności systemu wyszukiwania informacji są wskaźniki kompletności (stosunek liczby dokumentów wyszukanych i relewantnych do liczby dokumentów relewantnych w zbiorze) i dokładność (stosunek liczby dokumentów wyszukanych i relewantnych do liczby dokumentów wyszukanych), to wskaźniki te nie zależą w istotny sposób od jedności orzeczeń o relewancji, nawet w przypadkach, gdy jedność decyzji o relewancji sięga zaledwie 30%. Przeprowadzone badania eksperymentalne przez M.E. Lesk i G. Saltona [88] wykazały, że chociaż zbiory dokumentów są różnie zdefiniowane przy różnych orzekających o relewancji, to jednak w przypadku dokumentów najbardziej zbliżonych do kwerend orzeczenia w większości są jednomyślne. Ponieważ są to właśnie te dokumenty, które mają największy wpływ na wyznaczanie efektywności wyszukiwania, więc wskaźniki efektywności nie zależą w istotny sposób od jedności ocen.

W dalszej części cytowanej pracy autorzy przeprowadzili teoretyczne rozważania, w których za pomocą statystycznych wyliczeń wykazali, że warunki nałożone na orzeczenia o relewancji są bardzo słabe. Wymaga się tylko, aby orzekający z większym prawdopodobieństwem zakwalifikował dokument relewantny jako "relewantny" niż "relewantnym" nazwał dokument nierelewantny. Można przypuszczać, że różne zbiory ocen relewancji nie będą istotnie wpływały na inne miary efektywności systemu.

9. REDUNDANCJA W STRUKTURACH WYSZUKIWAWCZYCH

Metoda list inwersyjnych w rozproszonych systemach wyszukiwania informacji z określoną relacją hierarchii wprowadza bardzo dużą redundancję w strukturach wyszukiwawczych. Wprowadzenie wag deskryptorów zwiększa redundancję dwukrotnie, ponieważ w listach inwersyjnych adresy dokumentów występują razem z wagami. W czasie wyszukiwania dokumentów nie wszystkie adresy dokumentów z małymi wagami są wykorzystywane. Ta własność jest podstawą podanej w tym rozdziale metody zmniejszania redundancji dokumentów.

9.1.Redundancja a efektywność systemu

Metoda list inwersyjnych w systemach wyszukiwania informacji z określoną relacją uogólniania R na zbiorze deskryptorów wprowadza dużą redundancję względną dokumentów HR .

$$HR = 2 \left[\sum_{j=1}^m \sum_{t \in T_{j0}} \text{card}(L_j(t)) - m \right],$$

gdzie m jest liczbą podsystemów. We wzorze tym występuje czynnik 2 przed znakami sum, ponieważ listy inwersyjne zawierają pary postaci (adres dokumentu, waga).

W praktyce posługujemy się bezwzględną redundancją dokumentów BR , która wskazuje na średnią krotność powtórzeń dokumentów w listach inwersyjnych ($BR=HR/n$). W przykładowym badanym systemie redundancja względna WR dla $n=100$ dokumentów wynosiła $2*226$, bezwzględna zaś redundancja BR wynosiła $2*2.26$.

Podamy teraz jeden ze sposobów zmniejszania redundancji dokumentów. Zmniejszanie redundancji odbędzie się kosztem zmniejszania liczby dokumentów w niektórych odpowiedziach. Można założyć, że użytkownicy w swoich kwerendach dla ogólniejszych deskryptorów będą używać większych wag i żądać większych dokładności odpowiedzi niż dla deskryptorów szczegółowych. Obciążenia list inwersyjnych $b=cw$ (w - waga deskryptora

kwerendy, c - wymagana dokładność odpowiedzi) w większości przypadków dla ogólniejszych deskryptorów będą większe od obcięć list dla deskryptorów szczegółowych. Dlatego też z list inwersyjnych dla ogólniejszych deskryptorów adresy dokumentów z małymi wagami będą rzadziej wybierane niż adresy z większymi wagami. Adresy dokumentów z małymi wagami rzadko używane niepotrzebnie zwiększają redundancję dokumentów. Podamy teraz kryterium statystyczne wyboru wartości progowych b_k dla list inwersyjnych, z których można usunąć rzadko używane adresy dokumentów o wagach mniejszych od wartości progowych b_k .

Jeżeli obcięcia list $b=w*c$ będą mniejsze od wartości progowych b_k , to do odpowiedzi będą należeć tylko te adresy, których wagi są większe od b_k . W odpowiedzi będzie brakować adresów dokumentów, których wagi są z przedziału liczbowego (b, b_k) . W zależności od przyjętych parametrów systemu, przykładowo 1% lub 5% ogólniej liczby odwołań do listy inwersyjnej, wartości $b=w*c$ będą mniejsze od ustalonej wartości progowej b_k . Można więc przypuścić, że usunięcie adresów o wagach mniejszych od wartości progowej b_k nie spowoduje zbyt dużego zmniejszenia efektywności systemu, zwłaszcza że wśród niepełnych odpowiedzi brakować może tylko dokumentów o małych wagach deskryptorów.

Gdy liczba odwołań do danej listy w poszczególnych podsystemach w okresie między jedną a drugą reorganizacją jest zbyt mała, aby wyznaczyć wartość progową, wtedy listę zostawiamy bez zmian. Między reorganizacjami list inwersyjnych zbieramy informacje o liczbach b . W czasie reorganizacji systemu badamy rozkład liczb b . W algorytmie wyboru wartości progowej liczb b korzystamy z rozkładu beta [31].

Do empirycznych zdarzeń losowych zastosowano rozkład beta, ponieważ:

- (1) funkcja gęstość rozkładu beta jest określona na odcinku $[0,1]$.
- (2) funkcja gęstości może być zarówno symetryczna, jak i asymetryczna w zależności o parametrów p i q .

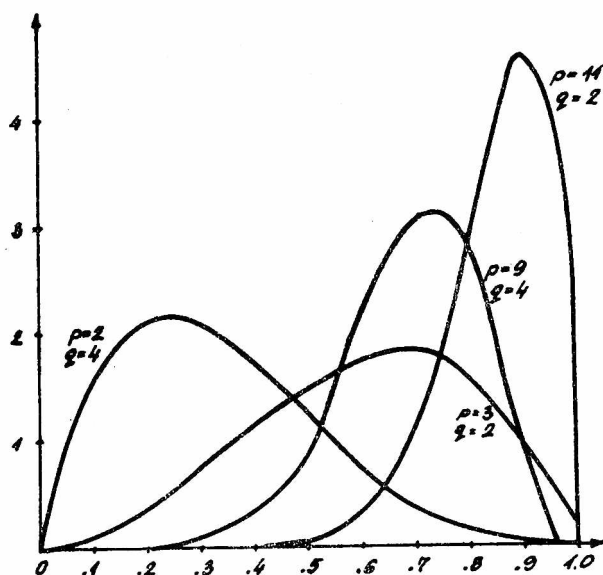
(3) bardzo prosto oblicza się parametry p i q , które są funkcjami pierwszego i drugiego momentu zwykłego m_1 , m_2 .

Funkcja gęstość rozkładu beta ma postać

$$f(x) = \begin{cases} \frac{x^p (1-x)^{q-1}}{B(p,q)} & 0 < x < 1 \\ 0 & x \leq 0 \wedge x \geq 1 \end{cases}$$

$$\text{gdzie } B(p,q) = \frac{\Gamma(p) \Gamma(q)}{\Gamma(p+q)}$$

Wykres funkcji gęstości w zależności od parametrów p , q przedstawiono na rys. 18.



Rys. 18. Wykresy funkcji gęstości rozkładu beta
Fig. 18. Representation of the density functions of the beta distribution

Funkcja gęstości wartość maksymalną przyjmuje w punkcie $(p-1)/(p+q-2)$.

Pierwszy i drugi moment zwykły rozkładu beta mają postać odpowiednio:

$$m_1 = p/(p+q) \quad (47)$$

$$m_2 = \frac{p(p+1)}{(p+q)(p+q+1)} \quad (48)$$

Rozwiązując układ dwóch równań (47), (48) ze względu na niewiadome p , q , otrzymujemy

$$p = m_1 \frac{m_1 - m_2}{m_2 - m_1} \quad (49)$$

$$q = (1 - m_1) \frac{m_1 - m_2}{m_2 - m_1} \quad (50)$$

9.2. Algorytm wyboru wartości progowych

Odcinek $[0,1]$ dzielimy na k podprzedziałów $(x_{i-1}, x_i]$ długości $1/k$, $(x_i = x_{i-1} + 1/k)$. Zbieranie informacji o wielkościach liczb $b=w*c$ przebiega

następująco: w każdym podsystemie S z listą $L_j(t)$ związany jest wektor P_t o k elementach. Jeśli $b \in (x_{i-1}, x_i]$, to $P_t(i) := P_t(i) + 1$. Po N kwerendach odwołujących się do listy inwersyjnej $L_j(t)$ elementy wektora $P_t(i)$ zawierają krotności (liczebność) liczb b z przedziału $(x_{i-1}/k, x_i)$, dla $i=1, 2, \dots, k$.

Elementy wektora P_t są wartościami funkcji rozkładu częstości $q(X)$ zmiennej losowej X podającymi częstość występowania i -tej realizacji tej zmiennej

$$q\{x_{i-1}/k < X < x_i\} = n_i = P_t(i),$$

przy czym:

$$(i) \quad n_i \geq 0,$$

$$(ii) \quad \sum_{i=1}^n n_i = N.$$

X jest zmienną losową typu skokowego o punktach skokowych x_i i skokach równych n_i/N . Pierwszy moment zmiennej losowej X przyjmuje wartość

$$m_1 = N \sum_{i=1}^k x_i \cdot n_i,$$

a drugi moment

$$m_2 = N \sum_{i=1}^k x_i^2 \cdot n_i.$$

Rozkład częstości można przedstawić za pomocą histogramu. Zmienna rozkładu, której rozkład można przedstawić w postaci histogramu, jest ciągła. Zwiększając nieograniczenie liczbę podprzedziałów na odcinku $[0, 1]$, otrzymujemy krzywą liczebności.

Wartość progową b_k znajdujemy rozwiązując nierówność

$$P(X < b_k) = \int_0^{b_k} \frac{1}{B(p, q)} x^{p-1} (1-x)^{q-1} dx \leq \alpha$$

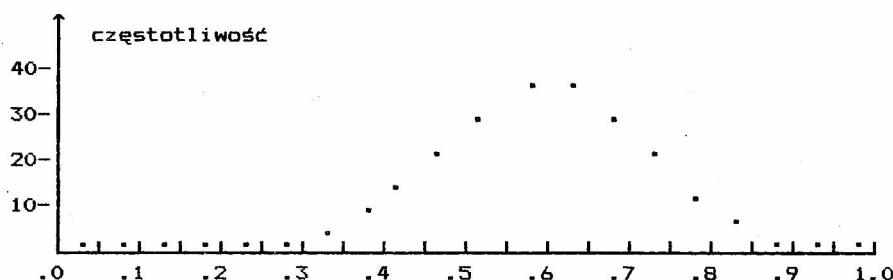
ze względu na niewiadomą b_k , gdzie α jest prawdopodobieństwem popełnienia błędu.

Przykład 18

Częstotliwość wystąpień liczb $b=c*w$ w poszczególnych przedziałach $(x_{i-1}/k, x_i]$ dla $k=20$ przedstawia się następująco:

i	6	7	8	9	10	11	12	13	14	15	16	17	18
n_i	1	3	7	13	21	29	34	34	28	19	10	3	1

Rozkład przedstawiony za pomocą histogramu ma postać



Obliczając momenty rozkładu m_1, m_2 ze wzorów (47), (48), otrzymujemy

$$m_1 = \frac{1}{N} \sum_{i=1}^{20} x_i \cdot n_i = 0.612,$$

$$m_2 = \frac{1}{N} \sum_{i=1}^{20} x_i^2 \cdot n_i = 0.387.$$

Korzystając ze wzorów (49), (50), obliczamy p i q ; $p=11.055$, $q=7.0086$. Odczytując z tabeli [110, s.133, 137] wartość progową dla $p=11$, $q=7$ i $\alpha=0.1$, otrzymujemy $b_k=0.343$. Dla $\alpha=0.05$ wartość b_k wynosi 0.421.

Z prawdopodobieństwem popełnienia błędu $\alpha=0.05$ z listy możemy usunąć adresy tych dokumentów, których wagi są mniejsze od 0.4 (gdyż wartość kryterialna na poziomie istotności $p=0.05$ wynosi 0.421). Do odpowiedzi nie będą należeć dokumenty, których wagi w listach są z przedziału (b, b_k) , przy tym 5% z ogólnej liczby odwołań do tej listy będzie spełniało warunek $b < b_k$.

W czasie wyszukiwania dokumentów z pamięci zewnętrznych będą sprowadzane listy inwersyjne do pamięci operacyjnej. Zmniejszenie redundancji dokumentów może wpłynąć nie tylko na zmniejszenie obszaru pamięci operacyjnej, ale też może przynieść korzyści użytkownikom w postaci zmniejszenia kosztów korzystania z systemu.

Gdyby w praktyce okazało się, że inne krzywe teoretyczne rozkładów prawdopodobieństwa lepiej przybliżają krzywą empiryczną na odcinku $[0,1]$, wtedy sposób wyznaczania wartości progowych dla każdej z list pozostanie taki sam.

Wyższym wartościom progowym w poszczególnych listach odpowiada

zmniejszenie redundancji dokumentów. W zależności od potrzeb użytkowników można odpowiednio sterować wartościami progowymi z każdej z list.

Jeśli w rozproszonym systemie wszystkie podzbiory dokumentów są sobie równe, tzn. $\forall_{1 \leq j \leq m} D_j = D$ oraz każdy deskryptor występuje dokładnie jeden raz w dowolnym podsystemie, to redundancja w takim systemie jest taka sama jak w scentralizowanym systemie. Jeśli natomiast w rozproszonym systemie jest spełniony warunek $\forall_{1 \leq j \leq m} D_j = D$, to lokalne listy inwersyjne dla tych samych deskryptorów są jednakowe, tj. $L_i(t) = L_k(t)$. Wtedy takie listy można przechowywać tylko w jednym z wybranych podsystemów.

10. KOSZTY KWEREND

Oceniając koszty kweryndy (koszty uzyskania odpowiedzi) bierzemy pod uwagę zarówno koszty obliczeń w węzłach, jak i koszty przesyłania danych. Czas pracy algorytmu wyszukiwania informacji, wyrażony jako funkcja rozmiaru problemu, nazywamy złożonością czasową algorytmu. Złożoność czasowa oznacza liczbę jednostek czasu potrzebnych do uzyskania odpowiedzi ze zbioru dokumentów rozmiaru n [1]. Dla ustalonego rozmiaru zbioru dokumentów można określić złożoność najgorszego przypadku (złożoność pesymistyczną) lub średnią złożoność dla wszystkich danych ustalonego rozmiaru (złożoność oczekiwaną). Złożoność średnią jest z reguły znacznie trudniej zanalizować niż złożoność pesymistyczną. Przede wszystkim należy przyjąć pewne założenia o rozkładzie danych. Założenia, które byłyby zbliżone do rzeczywistości powodują, że cały problem staje się bardzo skomplikowany.

W rozdziale tym będziemy się zatem koncentrować na analizie złożoności pesymistycznej, która jest bardziej uniwersalna. Jednak należy pamiętać, że algorytm o najlepszej złożoności pesymistycznej nie musi mieć najlepszej złożoności średniej.

Oznaczmy przez $C'_j(n_j, k)$ koszt kweryndy w podsystemie $S_j = (T_j, D_j, Q_j, f_j)$, gdzie k jest liczbą deskryptorów w kweryndzie $q \in Q_j$, $s_j = \text{card}(T_{0j})$, $n_j = \text{card}(D_j)$, $1 \leq j \leq m$. Zakładamy, że wymagana dokładność odpowiedzi wynosi $c=0$ (w trakcie wyszukiwania informacji będą wykorzystywane pełne listy inwersyjne).

Twierdzenie 16

$$C'_j(n_j, k) = O(n_j k \log(k) + (n+1)s_j \log(s_j)).$$

Dowód

W lokalnym podsystemie S_j odpowiedź na kweryndę

$q = \{(t_1, w_1), (t_2, w_2), \dots, (t_k, w_k)\}$ ma postać (14)

$$f_j(q, c) = \bigcap_{i=1}^k \left[LR(t_i) \Big|_{cw_i} \right].$$

Zgodnie z definicją 1, operacja "sumy" $\cup$ list inwersyjnych jest złożeniem operacji sumy mnogościowej $\cup$ oraz operacji zastąpienia wszystkich elementów list o takim samym numerze dokumentu, parą składającą się z tego dokumentu i wagi równej sumie wag, powtórzonych dokumentów.

Ponieważ w sumie list $LR(t_i) = \bigcup_{t \in X_i} L_j(t)$ liczba składników jest równa $\text{card}(X_i)$ i liczba elementów w $LR(t_i)$ jest mniejsza lub równa n_j , więc koszt utworzenia k list $LR(t_i)$ wynosi

$$\sum_{i=1}^k \log(\text{card}(X_i)) \text{card}(X_i) n_j \leq n_j \log(s_j) \sum_{i=1}^k \text{card}(X_i) \leq n_j s_j \log(s_j).$$

Ostatnia nierówność wynika z zależności

$$X_i \cap X_{i'} = \emptyset \Rightarrow \sum_{i=1}^k \text{card}(X_i) \leq \text{card}(s_j).$$

Koszt przecięcia k list inwersyjnych, każdej o długości co najwyżej n_j , wynosi $\log(k) \sum_{i=1}^k n_j = kn_j \log(k)$. Koszt odszukania wszystkich list inwersyjnych $L_j(t)$, których deskryptory są ściślejsze od deskryptorów kwerendy, wynosi $\sum_{i=1}^k \text{card}(X_i) \log(\text{card}(X_i)) \leq s_j \log(s_j)$.

Sumując koszty poszczególnych operacji, otrzymujemy $kn_j \log(k) + (n_j+1) \log(s_j)$, stąd pesymistyczny koszt kwerendy $O(kn_j \log(k) + (n_j+1)s_j \log(s_j))$.

■

Podobnie można obliczyć koszty globalnych kwerend w rozproszonym systemie $S=(T,D,Q,f)$ korzystając z definicji odpowiedzi (20)

$$f(q,c) = \bigcap_{i=1}^k \left[\bigcup_{t \in X_i} \bigcup_{j=1}^m L_j(t) \right] \Big|_{cw_i}$$

Niech k będzie liczbą deskryptorów w kwerendzie $q \in Q$ i $s = \text{card}(T_0)$, $n = \text{card}(D)$. Ponadto zakładamy, że dokładność odpowiedzi $c=0$.

Twierdzenie 17

$$C(n,k,m) = nk \log(k) + (n+m)s \log(s) + smn \log(m). \quad (51)$$

Dowód

Koszt utworzenia listy $L(t) = \bigcup_{j=1}^m L_j(t)$ ograniczony jest przez

$\log(m) \sum_{j=1}^m n_j \leq m n \log(m)$, natomiast koszt utworzenia k list inwersyjnych

postaci $LR(t_i) = \bigcup_{t \in X_i} \bigcup_{j=1}^m L_j(t)$ wynosi

$$\sum_{i=1}^k \text{card}(X_i) n \log(\text{card}(X_i)) + m n \log(m) \sum_{i=1}^k \text{card}(X_i) \leq n s \log(s) + s m n \log(m).$$

Koszt odszukania wszystkich list inwersyjnych w n lokalnych podsystemach wynosi $m s_j \log(s_j) \leq m s \log(s)$, a koszt przecięcia k list inwersyjnych $k n \log(k)$. Sumując koszty poszczególnych operacji otrzymujemy

$$k n \log(k) + n s \log(s) + s m n \log(m) + m s \log(s) = n k \log(k) + (n+m) s \log(s) + s m n \log(m).$$

■

Łatwo zauważyć, że dla $m=1$, $C(n, k, m) = C'_j(n_j, k)$.

Koszt kwerendy (51) dotyczy globalnego systemu S , w którym wszystkie listy inwersyjne lokalnych podsystemów przechowywane są w jednym węźle. Koszt C nie uwzględnia kosztów transmisji danych pomiędzy węzłami.

O czasie uzyskania odpowiedzi decyduje nie tylko liczba wykonanych obliczeń, jak zakłada się w scentralizowanych systemach, lecz także liczba danych przesyłanych między komputerami. Pomiedzy węzłami stosuje się łącza służące do przesyłania danych i komunikatów. Gdy stosuje się obecną technologię, czas przesłania nawet krótkiego komunikatu między węzłami nie jest bez znaczenia, trwa około 0.1 sekundy [97], a szybkość z jaką dane mogą być przesyłane wynosi od kilkuset do 10000 bajtów na sekundę w zależności od typu łączu komunikacyjnych.

Narzut 0.1 sekundy na wysłanie komunikatu odzwierciedla czas, w którym dwa połączone procesory wykonują protokół komunikacyjny. Jest on konieczny, aby jeden węzeł wysyłając komunikat był pewien, że ten drugi otrzyma go i że będzie wiedział co z nim zrobić. Szybkość wysyłania danych jest ograniczona szybkością ich wysyłania lub odbierania przez procesor albo też szybkością, z którą dane mogą być przesłane łączem, bądź też oboma tymi wielkościami.

Koszty przesyłania list inwersyjnych z podsystemu S_j do globalnego systemu można wyrazić w postaci

$$C_P = \sum_{i=1}^k \sum_{t \in X_i} \sum_{j=1}^m \delta_1^j (\alpha + \beta \text{card}(L_j(t))) / \gamma \quad (52)$$

gdzie:

m - liczba lokalnych podsystemów,

k - liczba deskryptorów w kwerendzie,

X - zbiór deskryptorów ściślejszych od deskryptora t ,

$$\delta_i^j = \begin{cases} 1 & \text{jeśli } t \in T_j \wedge 1 \leq i \leq n \wedge 1 \leq j \leq n, \\ 0 & \text{w p.p.} \end{cases}$$

α - czas przesłania komunikatu między węzłami,

β - liczba bajtów potrzebnych do przechowywania jednego elementu listy,

γ - szybkość transmisji danych łączy sieci (bajt/s).

Zakładamy, że transmisje list ze wszystkich podsystemów odbywają się równocześnie (w przeciwnym przypadku koszt C_p należy pomnożyć przez liczbę węzłów).

Pesymistyczny czas przesłania list inwersyjnych z lokalnych podsystemów do globalnego systemu wynosi

$$C_{pp} = \text{card}(T_0) (\alpha + \beta n / \gamma).$$

Globalny koszt C kwerendy uwzględniający koszty obliczeń w węzłach i koszty przesyłania danych ma postać

$$C = C(n, k, m) \omega + C_{pp},$$

gdzie ω jest kosztem jednostkowym wykonania jednej operacji w węzle.

Przykład 19

Założmy, że rozproszony system S składa się z $m=4$ podsystemów oraz że kwerenda q zawiera $k=4$ deskryptory z wagami, takie że dla każdego z nich w tezaurusie T_0 istnieją po 3 deskryptory szczegółowsze.

Aby utworzyć globalną odpowiedź, należy wykorzystać $mk(1+3)=64$ listy inwersyjne z lokalnych podsystemów S_j ($1 \leq j \leq 4$). Założmy ponadto, że wszystkie lokalne listy inwersyjne są jednakowej długości i zawierają po n elementów.

Niech $\alpha=0.1$, $\beta=4$, $\gamma=5000$ bajtów/s, $\omega=10^{-6}$ s.

Koszty obliczeń w węzłach $C(n, k, m)$, koszty C_{pp} przesłania 64 list inwersyjnych z lokalnych podsystemów do globalnego systemu oraz globalne koszty kwerendy przedstawiają się następująco:

Koszty	$n=10^2$	$n=10^3$	$n=10^4$
C_{pp}	1.68	2.4	9.6
$C(n, k, m)$	0.09	0.9	9.0
C	1.77	2.2	18.6

Koszty "średnie" kwerend są trudne do obliczenia analitycznie. Zależą one m.in. od

(a) Wymaganej dokładności odpowiedzi. Przy większych dokładnościach czas wykonania kwerendy będzie krótszy niż przy małych dokładnościach, ponieważ do przygotowania odpowiedzi będą wykorzystywane obcięte listy. Obcięcie listy zależy od dokładności odpowiedzi.

(b) Postaci kwerend. Koszty kwerend ogólniejszych będą większe od kosztów kwerend szczegółowych.

(c) Postaci tezaurusów. Koszty kwerend będą większe, gdy wszystkie deskryptory globalnego tezaurusu będą występować w każdym lokalnym tezaurusie, natomiast gdy deskryptory te będą występować w pojedynczych podsystemach, wtedy koszty kwerend będą mniejsze. Podobna zależność dotyczy zbioru dokumentów.

(d) Algorytmu sterowania współbieżnością transmisji lokalnych list inwersyjnych.

Porównując pesymistyczne koszty kwerend $C(n,k,m)$ w rozproszonych systemach i $C'_j(n,k)$ w scentralizowanych systemach można zauważyć, że różnica kosztów obliczeń w węzłach wynosi $C(n,k,m) - C'_j(n,k) = ms \log(s) + nms \log(m)$.

Przykład 20

Założmy, że wszystkie lokalne podsystemy zawierają jednakowe tezauryusy i te same zbiory dokumentów. Dla danych podanych w przykładzie 10 liczba wykonanych obliczeń w rozproszonym systemie zwiększyła się 2.3-krotnie w porównaniu z obliczeniami w scentralizowanym systemie.

Koszty kwerend zwiększają się, jeśli uwzględnia się koszty transmisji w sieci. Prędkość transmisji danych jest znacznie mniejsza niż prędkość, z którą można odczytać listę z dysku. Przekazywanie danych między komputerami jest więc wąskim gardłem i większość zagadnień specyficznych dla systemów rozproszonych [53] dotyczy sposobów uporania się z tym problemem [91], [97].

W czasie eksploatacji systemu oceny użytkowników o relewancji dokumentów należących do odpowiedzi są gromadzone w postaci zmodyfikowanych opisów dokumentów. Po pewnym okresie pracy systemu niektóre deskryptory mogą stracić swoją poprzednią ważność w opisach dokumentów poprzez zmianę wag. Stara postać list inwersyjnych po pewnym czasie nie odzwierciedla nowych postaci opisów dokumentów. Dlatego też co pewien czas należy aktualizować listy inwersyjne. Ze zmodyfikowanych

opisów dokumentów użytkownicy będą mogli skorzystać dopiero po reorganizacji list, ponieważ w czasie modyfikacji opisów dokumentów listy inwersyjne pozostają bez zmian. Koszty modyfikacji opisów dokumentów nie wpływają na koszty kwerend, ponieważ modyfikacje możemy traktować jako modyfikację zbiorów w lokalnych podsystemach.

Odpowiedzi na kwerendy najczęściej zawierają niewiele elementów ze zbioru dokumentów, stąd koszty modyfikacji opisów dokumentów nie będą duże. Na koszt reorganizacji list inwersyjnych składa się koszt jednokrotnego przeglądu zbiorów dokumentów i dopisania tych dokumentów do odpowiednich list inwersyjnych. Jeśli wszystkie lokalne tezaury i zbiory dokumentów są sobie równe, to koszt modyfikacji opisów będzie równy sumie kosztów modyfikacji w lokalnych podsystemach.

Podstawową grupę użytkowników rozproszonych systemów wyszukiwania informacji będą stanowić osoby pragnące korzystać ze zbiorów danych przechowywanych w różnych komputerach. Z praktyki wiadomo, że większość lokalnych zasobów programowych i baz danych była tworzona niezależnie od siebie. Powstawało oprogramowanie różnorodne pod względem organizacyjnym i metod organizacji lokalnych baz danych.

Rozproszone systemy umożliwiają dostęp do lokalnych zbiorów dokumentów w komputerach pracujących w sieci. W rozproszonych systemach pojawia się nowa grupa użytkowników globalnych mająca dostęp do lokalnych zasobów. Lokalni użytkownicy korzystają z podsystemów na dotychczasowych zasadach oraz udostępniają swoje zasoby globalnym użytkownikom.

W wielu przypadkach podział dokumentów jest naturalny. Każda instytucja może mieć własne zasoby informacyjne dotyczące danych bibliograficznych, personelu, wyrobów itp. W większości programów użytkowych w danej instytucji korzysta się z lokalnych zbiorów danych tej instytucji, lecz w wielu sytuacjach potrzebne są dane z innego lub innych zasobów informacyjnych. Ponadto bywają również przypadki, że zasoby danych są rozdzielone nie w sposób "naturalny", lecz użytkownik dla większej pewności przechowuje dane w dwóch lub więcej miejscach (np. banki, wojsko, itp.).

Jeszcze inną przyczyną rozdziału zasobów informacyjnych jest ich wielkość. Nie można przyłączyć do jednego komputera dowolnie dużej liczby jednostek dyskowych. Powyżej pewnej granicy użytkownik jest wręcz zmuszony do zastosowania rozproszonego systemu. Poza tym, aby móc w sposób elastyczny rozbudowywać system, lepiej jest dysponować wieloma mniejszymi instalacjami niż jedną dużą.

Uwzględniając powyższe realia oraz rozwój sieci komputerowych, można stwierdzić, że są one argumentami potwierdzającymi celowość dalszego rozwoju koncepcji przetwarzania rozproszonego.

11. ZAKOŃCZENIE

Rozważania i otrzymane wyniki symulacji procesu wyszukiwania sugerują, że podane metody modyfikacji opisów dokumentów w rozproszonej przestrzeni opisów dokumentów mogą być pomocne w zwiększeniu efektywności rozproszonych systemów wyszukiwania informacji. Oczywiście jest, że metody takie przynoszą korzyści zarówno z punktu widzenia efektywności wyszukiwania, jak i indeksowania oraz że korzyści te wiążą się z dodatkowymi kosztami wyszukiwania.

Otrzymane wyniki mogą być podstawą do dalszych badań i zastosowań praktycznych.

Istniejące systemy wyszukiwania informacji oparte na tezaursach można niezbyt dużymi kosztami przystosować do warunków modelu podanych w rozdziale 2 i włączyć do rozproszonego systemu. Wystarczy np. wszystkim deskryptorom w opisach dokumentów nadać początkowe wagi równe $1/f_j$ (f_j - liczba różnych deskryptorów występujących w j -tym opisie dokumentu). Listy inwersyjne, spełniające założenia z rozdziału 1.4, można otrzymać dokonując jednego pełnego przeglądu wszystkich opisów dokumentów. W ten sposób otrzymamy rzeczywisty system spełniający założenia badanego systemu, w którym możemy prowadzić dalsze badania i korzystać z systemu w celach praktycznych.

Ze zmodyfikowanej przestrzeni dokumentów korzyści czerpać będą nie tylko ci użytkownicy, którzy brali udział w modyfikowaniu opisów dokumentów. Nowi globalni i lokalni użytkownicy korzystać będą z "doświadczeń" zebranych podczas poprzednich wyszukiwań.

Gromadzenie i wyszukiwanie informacji pochodzących z różnych źródeł i udostępnianie ich wielu użytkownikom oraz reprezentacja złożonych struktur danych w postaci wielu plików danych prowadzi również do dużych trudności w utrzymaniu logicznej spójności danych. Coraz większe znaczenie zasobów informacyjnych i złożoność ich struktur powodują konieczność tworzenia rozproszonych systemów, których podstawową funkcją jest zarządzanie tymi zasobami.

Analiza mechanizmów algorytmów wyszukiwania informacji umożliwia dokonywanie opisów powiązań poszczególnych elementów danych w sposób niezależny od stosowanych technik ich zapisu na nośnikach urządzeń pamięci zewnętrznych oraz badanie struktur logicznych metod wyszukiwania

informacji niezależnie od problemów związanych z komunikacją między poszczególnymi komputerami. Znając dostatecznie te mechanizmy, możemy być pewni, że wyszukane odpowiedzi będą prawidłowe i że będą one wyszukiwane w optymalny sposób.

Implementując rozproszone systemy wyszukiwania informacji, należy pamiętać, aby z punktu widzenia użytkownika końcowego rozproszony system działał jako scentralizowany system. Użytkownik końcowy nie powinien ani zdawać sobie sprawy z miejsca przetwarzania i przechowywania danych, ani umieć je wskazać. Natomiast z punktu widzenia projektanta systemu należy uważać system za rozproszony, gdy wszystkie decyzje co do miejsca przetwarzania i przechowywania podejmowane są przez sam system [53].

Ostatnie badania w dziedzinie modelowania danych są skierowane na normalizację pojęć i porównywanie modeli, matematyczną formalizację modeli w celu lepszego rozeznania ich ograniczeń oraz wykorzystania koncepcji z dziedziny sztucznej inteligencji do budowy nowych modeli lepiej oddających semantykę świata rzeczywistego.

Najważniejsze problemy, które należy rozwiązać w przyszłości to:

(a) Budowa prawdziwie rozproszonego systemu. Istnienie rzeczywistego systemu pozwoliłoby ocenić nowe wyniki teoretycznych badań z zakresu sterowania równocześnieścią, metod dostępu do danych, dekompozycji kwerend i przetwarzania częściowych odpowiedzi z lokalnych podsystemów.

(b) Prowadzenie dalszych prac teoretycznych nad nowymi modelami rozproszonych systemów,

(c) Zbadanie możliwości współpracy istniejących systemów wyszukiwania informacji opartych na różnych modelach,

(d) Przeprowadzenie analizy porównawczej struktur informacyjnych i metod indeksowania w różnych systemach pod kątem włączenia ich (jako lokalnych podsystemów) do systemu rozproszonego.

Systemy adaptacyjne nabiorą większego znaczenia w przypadku rozproszonym. Jednym z podstawowych celów adaptacyjnych może być "dostrojenie" lokalnych podsystemów do współpracy z systemem globalnym.

Praca wpłynęła do redakcji 23.08.1988 r.

Po poprawkach 8.05.1989 r.

LITERATURA

- [1] Aho A.V., Hopcroft J.E., Ullman J.D.: *Projektowanie i analiza algorytmów komputerowych*. Warszawa, PWN, 1983. *The Design and Analysis of Computer Algorithms*. Addison-Vesley Publishing Company, 1974.
- [2] Bärtschi M.: *An overview of information retrieval subject*. IEEE Computer, 18, no. 5, 1985, s. 67-84.
- [3] Belkin M. i in.: *Distributed expert-based information system. An interdisciplinary approach*. Information Processing & Management, 23, no. 5, 1987, s. 395-409.
- [4] Biswas i in.: *Knowledge - assisted document retrieval: I. The natural-language interface*. Journal of the American Society for Information Science, 38, 1987, s. 83-96.
- [5] Bolc L., Cichy M., Różańska L.: *Przetwarzanie języka naturalnego*, Warszawa, WNT, 1982.
- [6] Bookstein A.: *Explanation and Generalization of Vector Models in Information Retrieval*. W: Salton G., Schneider H.J.: *Research and Development in Information Retrieval. Proc. Berlin 1982, Lecture Notes in Computer Science*, vol. 146, Berlin, Springer-Verlag, 1983.
- [7] Bookstein A.: *Recent Developments in the theory of information retrieval. Report TRITA-LIB-6019*, Royal Institute of Technology Library, Sztokholm, 1982.
- [8] Bookstein A.: *A comparison of two systems of weighted Boolean retrieval*. Journal of the ASIS, 32, no. 4, July 1981.
- [9] Bookstein A.: *Fuzzy requests: An approach to weighted Boolean searches*. Journal of the American Society for Information Science, July 1980, s. 240-247.
- [10] Bookstein A.: *On the perils of merging Boolean and weighted retrieval systems*, Journal of the ASIS, 29, 1978.
- [11] Bookstein A., Swanson D.R.: *A Decision Theoretic Foundation for Indexing*. Journal of the ASIS, vol. 26, no. 1, 1975.
- [12] Bookstein A., Swanson D.R.: *Probabilistic Models for Automatic Indexing*. Journal of the ASIS, vol. 25, no. 5, 1974.
- [13] Buell D.: *A problem in information retrieval with fuzzy sets*. Journal of the American Society for Information Science, 36, no. 6, 1985, s. 398-401.
- [14] Buell A.: *An analysis of some fuzzy subset applications to information retrieval systems*. Fuzzy Sets and Systems, 7, 1982, s. 35-42.
- [15] Buell D.A.: *A general model of query processing in information retrieval systems*. Information Processing & Management, 17, no. 5, 1981, s. 249-262.
- [16] Buell D.A., Kraft D.K.: *Performance measurement in a fuzzy retrieval environment*. Proceedings of the Fourth International

- Conference on Information Storage and Retrieval, Oakland, California, maj 31 - czerwiec 2, 1981.
- [17] Chang Y.K., Cirillo C., Razon J.: *Evaluation of feedback retrieval using modified freezing, residual collection, and test and control groups*. W: The SMART System - Experiments in Automatic Document Processing, Englewood Cliffs, NJ, Prentice-Hall, chapt. 17, 1971.
- [18] Choroś K., Daniłowicz Cz.: *Relative indexing: weighted descriptors and relative indexing in a document retrieval system model*. Information Processing & Management, 18, 1982, s. 207-220.
- [19] Chu W.W.: *Principles of Distributed Data-Base Design*. W: *Computer Communications*, vol. II, Systems and Applications. Prentice-Hall, 1983, s. 434-480.
- [20] Cooper W.S., Maron M.E.: *Foundations of probabilistic and utility-theretic indexing*. Journal of the Association for Computing Machinery, 25, 1978, s. 67-80.
- [21] Croft W.B.: *Boolean Queries and term dependencies in probabilistic retrieval models*. Journal of the American Society for Information Science, 37, no. 2, 1986, s. 71-77.
- [22] Croft W.B.: *Experiments with representation in a document retrieval system*. Information Technology: Research and Development, no. 2, 1983, s. 1-21.
- [23] Croft W.B.: *Document Representation in Probabilistic Models of Information Retrieval*. Journal of the ASIS, vol. 32, 1981.
- [24] Croft W.B.: *A model of cluster searching based on classification*. Information systems, 5, 1980, s. 189-195.
- [25] Croft W.B., Harper D.J.: *Using probabilistic models of document retrieval without relevance information*. Journal of Documentation, 4, no. 4, 1979, s. 285-295.
- [26] Czerny A.I.: *Teoria wyszukiwania informacji*. Warszawa, PWN, 1981. Введение в теорию информационного поиска. Издательство "Наука", Москва, 1975.
- [27] Date C.J.: *An Introduction to Database Systems*, Addison-Wesley Publishing Company, 1986.
- [28] Dąbrowski M.: *Modele systemów wyszukiwania informacji*. Prace Instytutu Organizacji i Kierowania PAN, seria B, z. 16, 1975.
- [28] Dąbrowski M., Laus K.: *A model of information retrieval processes for hierarchical set of descriptors*. Information Storage & Retrieval, 10, 1974, s. 261-265.
- [30] Dąbrowski M., Laus-Maczyńska K.: *Metody wyszukiwania i klasyfikacji informacji*. Warszawa, WNT, 1978.
- [31] Fisz M.: *Rachunek prawdopodobieństwa i statystyka matematyczna*. Warszawa, PWN, 1967.
- [32] Griffiths A., Luckhurst C., Willet P.: *Using interdocument similarity information in document retrieval systems*. Journal of the American Society for Information Science, 37, 1986, s. 3-11.
- [33] Hall H.: *Combinatorial Theory*. Waltham Massachusetts, 1967.
- [34] Harter S.P.: *A probabilistic approach to automatic keyword indexing*. Part I: *On the distribution of speciality words on a technical literature*, Part II: *On algorithms for probabilistic inferencing*. Journal Am. Soc. Inf. Sci., 26, 1975.
- [35] Ide E.: *New experiments in relevance feedback*. W: *The SMART System - Experiments in Automatic Document Processing*. Englewood Cliffs, NJ, Prentice-Hall. chapt.16, 1971.

- [36] Ide E., Salton G.: *Interactive search strategies and dynamic file organization in information retrieval*. W: *The SMART System - Experiments in Automatic Document Processing*, Englewood Cliffs, NJ, Prentice-Hall, chapt. 18, 1971.
- [37] Kamiński W., Rode-Babczenko I.: *STAIRS - system wyszukiwania informacji*. Informatyka, nr 10, 1975.
- [38] Keen E.M.: *Analiza kwerend z dziedziny dokumentacji*, W: *The SMART retrieval systems*. Prentice-Hall, 1971, tłum.pol. *SMART automatyczny system wyszukiwania informacji*, WNT, Warszawa, 1975.
- [39] Kłyk J., Jurek J.K.: *Dialogowe semiotyczne systemy podejmowania decyzji*. Warszawa, PWN, 1988.
- [40] Krygier B.: *Wzrost liczby publikacji naukowych a systemy informacyjne*. W: *Materiały konferencyjne; Systemy Wyszukiwania Informacji*. Warszawa, PWN, 1974, s. 159-166.
- [41] Kubicz R.: *Własności rozproszonego systemu wyszukiwania informacji opartego na tezaursie z wagami*. Praca magisterska, Wydział Informatyki i Zarządzania PWr., 1987.
- [42] Later E.: *Własności modelu systemu wyszukiwania informacji opartego na tezaursie z wagami z uwzględnieniem deskryptorów zanegowanych*. Praca magisterska, Wydział Informatyki i Zarządzania PWr., 1983.
- [43] Leski K.: *Zasady budowy tezaursów*. Prace OIN PAN, Warszawa, 1978.
- [44] Łukasiewicz W.: *Formalization of knowledge and ignorance*. The Journal for the Integrated Study of Artificial Intelligence Cognitive Science and Applied Epistemology, 1986. Materiały Szkoły PTI Bazy Danych'85, Karpacz 1985, s. 1-31.
- [45] Macleod I. i in.: *Strategies for building distributed information retrieval system*. Information Processing & Management, 23, no. 6, 1987, s. 511-528.
- [46] Marek W.: *On the data base machine*. ICS PAS Report, 1979, nr 347, Warszawa.
- [47] Maron M.E., Kuhns J.L.: *On relevance, probabilistic indexing and information retrieval*. Journal of the Association for Computing Machinery, 3, 1960, s. 216-244.
- [48] Mazur Z.: *Properties of a model of distributed homogeneous information retrieval system based on weighted models*. Information Processing & Management, 24, no. 5, 1988, s. 525-540.
- [49] Mazur Z.: *Properties of a distributed information retrieval system based on descriptor sets*. Raporty Centrum Obliczeniowego Politechniki Wrocławskiej, ser. PRE, nr 10/88, 1988.
- [50] Mazur Z.: *Properties of a model of distributed homogeneous information retrieval system based on thesauri with weights*. Raporty Centrum Obliczeniowego PWr., ser. PRE, nr 8/87, 1987.
- [51] Mazur Z.: *Organization of the inverted files in a distributed information retrieval system based on thesauri*. Information Processing & Management, 22, 1986, s. 243-250.
- [52] Mazur Z.: *On a model of distributed information retrieval system based on thesauri with weights*. Referat wygłoszony na międzynarodowym seminarium "Bazy Danych" 29.01-1.02.1986 w Magdeburgu. Raporty Centrum Obliczeniowego Politechniki Wrocławskiej, ser. PRE, nr 3/86, 1986.
- [53] Mazur Z.: *Bazy danych w sieciach komputerowych*, Prace Naukowe Centrum Obliczeniowego Politechniki Wrocławskiej, Seria: Konferencje; Sieci Komputerowe, 1985, s. 81-96.

- [54] Mazur Z.: *On a model of distributed information retrieval system*. Wiss. Z. Techn. Hochsch., Magdeburg, 29, 1985, s. 56-58.
- [55] Mazur Z.: *Organization of the inverted files in a distributed information retrieval system*. Wiss. Z. Techn. Hochsch., Magdeburg, 24, 1985, s. 59-61.
- [56] Mazur Z.: *On a model of distributed information retrieval systems based on thesauri*. Information Processing & Management, 20, 1984, s. 499-505.
- [57] Mazur Z.: *Die Methode des Aufsuchens von Informationen mit Hilfe der auf der Menge von Deskriptoren mit Gewichten definierten hierarchischen Relation*. Wiss. Z. Techn. Hochsch. Magdeburg, 24, Heft 3, 1980, s. 51-52.
- [58] Mazur Z.: *Properties of a model of information retrieval system based on thesaurus with weights*. Information Processing & Management, 15, 1979, s. 145-154.
- [59] Mazur Z.: *Inverted file organization in the information retrieval system based on thesaurus with weights*. Information Processing & Management, 15, 1979, s. 227-234.
- [60] Mazur Z.: *Метод поиска информации учитывающий отношение иерархии на множестве дескрипторов с весами*. 5th Symposium on Algorithms. Algorithms'79, Vysoke Tatry, Strske Pleso, April 23-27, 1979, s.344-348.
- [61] Mazur Z.: *Własności i zastosowania modelu systemu wyszukiwania informacji opartego na tezaurusie z wagami*. Rozprawa doktorska, Uniwersytet Warszawski, 1977.
- [62] Moser J., Christoph R.: *Management expert system (M.E.S.): a framework for development and implementation*. Information Processing & Management, 23, no. 1, 1987, s. 17-23.
- [63] Noreault T., McGill M., Koll M.B.: *A performance evaluation of similarity measures, Document term weighting schemes and representation in a Boolean environment*. W: Ody R.N. (ed.): *Information Retrieval Research*, Butterworths, London, 1981.
- [64] Noreault T., Koll M., McGill M.J.: *Automatic Ranked Output from Boolean Searches in SIRE*. Journal of the ASIS, vol. 28, no. 6, 1977.
- [65] Ochman J. (red.): *Organizacja systemów rozproszonego przetwarzania danych*. Warszawa, PWE, 1987.
- [66] Pawlak Z.: *Distributed information systems*. ICS PAS Report, 370, 1979.
- [67] Pawlak Z.: *Information systems-theoretical foundation*. Information Systems, 6, no. 3, 1981, s. 205-208.
- [68] Pawlak Z.: *Systemy Informacyjne. Podstawy Teoretyczne*. Warszawa, WNT, 1983.
- [69] Raś Z.: *Information retrieval systems, an algebraic approach I*. Annales Societatis Mathematicae Polonae. Fundamenta Informaticae. 4, 1981, s. 551-604.
- [70] Radecki T.: *Mathematical model of information retrieval system based on the concept of fuzzy thesaurus*. Information Processing & Management, 12, 1976.
- [71] Raghavan V., Wong S.K.M.: *A critical of vector space model for information retrieval*. Journal of the American Society for Information Science, 37, no. 5, 1986, s. 279-287.
- [72] Raghavan V.V., YU C.T.: *Experiments on the determination of the*

- relationship between terms*. ACM Transaction on Database Systems, 4, no. 2, 1979.
- [73] Rocchio J.J.: *Relevance feedback in information retrieval*. W: *The SMART System - Experiments in Automatic Document Processing*. Englewood Cliffs, NJ, Prentice-Hall, chapt. 14, 1971.
- [74] Robertson S.C., Maron M.E., Cooper W.S.: *Probability of relevance: A unification of two competing models for document retrieval*. Information Technology : Research and Development, no. 1, 1982, s. 1-21.
- [75] Robertson S.E., Spark Jones K.: *Relevance weighting of search terms*. Journal of the American Society for Information Science. 27, 1976, s. 129-146.
- [76] Robertson S.E., Thompson C.L., Macaskil M.J.: *Weighting, ranking and relevance feedback in a front-end system*. Journal of Information Science, 12, 1986, s. 71-75.
- [77] Rousseau R.: *On relative indexing in fuzzy retrieval systems*. Information Processing & Management, 21, no. 5, 1985, s. 415-417.
- [78] Salton G.: *Another Look at Automatic Text Retrieval Systems*. Technical Report TR 85-713, Cornell University, 1985.
- [79] Salton G.: *Some Characteristics of Future Information Systems*. ACM SIGIR Forum, vol. 18, 1983, s. 28-39.
- [80] Salton G.: *Mathematics and Information Retrieval*. Journal of Documentation, vol. 35, no. 1, 1979.
- [81] Salton G.: *Dynamic Information and Library Processing*. Prentice-Hall, Englewood Cliffs, 1975.
- [82] Salton G.: *Relevance feedback and the optimization of retrieval effectiveness*. W: *The SMART System-Experiments in Automatic Document Processing*. Englewood Cliffs, NJ, Prentice-Hall, chapt.15, 1971.
- [83] Salton G.: *Automatic information organization and retrieval*, McGraw-Hill, New York, 1968.
- [84] Salton G., Buckley C., Fox E.A.: *Automatic Query Formulation in Information Retrieval*. Journal of the American Society for Information Science, 34, no. 4, 1983, s. 262-280.
- [85] Salton G., Fox E.A., Wu H.: *Extended Boolean information retrieval*. Communications of the ACM, 26, no. 11, 1983.
- [86] Salton G., Fox E.A., Voorhees E.: *Advanced feedback methods in information retrieval*. Journal of the American Society for Information Science, 36, no. 3, 1985, s. 200-210.
- [87] Salton G., McGill M.J.: *Introduction to modern information retrieval*. New York, McGraw-Hill, 1983.
- [88] Salton G. (ed): *The SMART Retrieval System, Experiments in Automatic Document Processing*. New Jersey, Prentice-Hall, Englewood Cliffs, 1971.
- [89] Subramanian V., Biswas G., Bezdek J.: *Document retrieval using a fuzzy knowledge-based system*. Optical Engineering, 25, no. 3, 1986, s. 445-455.
- [90] Tahani V.: *A fuzzy model of document retrieval systems*. Information Processing & Management. 12, 1976, s. 177-187.
- [91] Tanenbaunn A.: *Sieci komputerowe*. WNT, Warszawa, 1988. *Computer Networks*, New Jersey, Prentice-Hall Inc., 1981.
- [92] Travis I.L., Fidel R.: *Subject analysis*. Annual Review of Information Science and Technology, ASIS, White Plains, 1982.

- [93] Turski W.M.: *Propedeutyka informatyki*. wyd. 1, Warszawa, PWN, 1975.
- [94] Turski W.M.: *Niektóre zagadnienia struktur danych*. W: *Problemy przetwarzania informacji*, tom 2, Warszawa, WNT, 1974, s. 153-197.
- [95] Turski W.M.: *A correspondence between thesaurus-based and multi-attribute retrieval systems*. *Information Storage & Retrieval*, 8, 1972, s. 309-313.
- [96] Turski W.M.: *On a model of information retrieval system based on thesaurus*. *Information Storage & Retrieval*, 7, 1971, s. 88-94, 201.
- [97] Ullman J.D.: *Principles of Database Systems*. Stanford University, 1980. *Systemy baz danych*. Warszawa, PWN, 1985.
- [98] van Rijsbergen C.J.: *Information Retrieval*, Butterworths, London, 1979.
- [99] Willet P.: *Document clustering using an inverted file approach*. *Journal of Information Science*, 2, 1980, s. 223-231.
- [100] Willet P.: *Similarity coefficient and weighting for automatic document classification: an empirical comparison*. *Int. Classif.*, 10, no. 3, 1983, s. 138-142.
- [101] Willet P.: *Query-specific automatic document classification*. *Int. Forum Inf. and Doc.*, 10, no. 2, 1985, s. 28-32.
- [102] Wong S.K.M., Ziarko W.: *A unified model in information retrieval*. *Fundamenta Informaticae*, 10, 1987, s. 35-56.
- [103] Wong S.K.M., Ziarko W.: *A machine learning approach to information retrieval*. *ACM Conference on Research and Development in Information Retrieval*, 1986, s. 228-233.
- [104] Wong S.K.M., Ziarko W.: *Query processing in information retrieval systems based on the generalized vector space model*. *Proc. Fourth Hung. Computer Sci. Conf.* M.Arato, I.Katai, L.Varga (ed.), 1985, s. 399-410.
- [105] Wong S.K.M., Ziarko W.: *On generalized vector space model in information retrieval*. *Fundamenta Informaticae*, 4, 1985, s. 254-267.
- [106] Wong S.K.M., Ziarko W., Raghavan V.V., Wong P.C.N.: *On modeling of information retrieval concepts in vector spaces*. *ACM Transaction on Database Systems*, 12, June 1987, s. 299-321.
- [107] Yu C.T., Luk W.S., Siu M.K.: *On models of information retrieval process*. *Information Processing & Management*, 4, 1979, s. 205-218.
- [108] Yu C.T., Salton G., Siu M.K.: *Effective automatic indexing using term addition and deletion*, *JACM*, 25, 1978, s. 210-225.
- [109] Zadeh L.A.: *Fuzzy sets*. *Information and Control*, 8, 1965.
- [110] Zieliński R.: *Tablice statystyczne*, Warszawa, PWN, 1972.

DODATEK

TABELA 1. Współczynniki podobieństwa ρ dla odpowiedzi na 15 globalnych kwerend po modyfikacji w lokalnych podsystemach

Nr kwerendy	$\rho(X, Y)$	$\rho(X, Z)$
1.	0.9254	0.9735
2.	0.8667	0.8691
3.	0.9655	0.9962
4.	0.9412	0.9735
5.	0.9412	0.9507
6.	0.9333	0.9937
7.	0.9315	0.9432
8.	0.9231	0.9520
9.	0.9412	0.9574
10.	0.9091	0.9199
11.	0.7407	0.9718
12.	0.7234	0.9002
13.	0.6667	0.8857
14.	0.9730	0.9964
15.	0.9697	0.9772

$$\bar{\rho}(X, Y) = 0.890$$

$$\bar{\rho}(X, Z) = 0.953$$

TABELA 2. Współczynniki podobieństw ρ dla list inwersyjnych po modyfikacji opisów w lokalnych podsystemach

Podsystem S_1			Podsystem S_2		
nr listy	$\rho(X, Y)$	$\rho(X, Z)$	nr listy	$\rho(X, Y)$	$\rho(X, Z)$
1.	1.0000	1.0000	1.	1.0000	1.0000
2.	0.8333	0.8516	19.	0.8468	0.9077
3.	0.7711	0.8542	20.	0.8571	0.8914
4.	0.8182	0.8128	21.	0.7719	0.8623
5.	0.7887	0.7952	22.	0.7647	0.9144
6.	0.7536	0.7764	23.	0.6923	0.7297
7.	0.8750	0.8750	24.	0.7907	0.8241
8.	0.8376	0.8606	25.	0.7925	0.8376
9.	0.7500	0.7872	26.	0.8515	0.8831
10.	0.7333	0.7594	27.	0.8421	0.8491
11.	0.6667	0.6667	28.	0.8571	0.8744
12.	0.7500	0.8375	29.	0.8000	0.7907
13.	0.8000	0.8071	30.	0.8250	0.8124
14.	0.5714	0.5714	31.	0.8649	0.8706
15.	0.7273	0.7374	32.	0.8197	0.8222
16.	0.7862	0.8645	33.	0.8615	0.8968
17.	0.8125	0.8535	34.	0.8108	0.8387
18.	0.8267	0.8956	35.	0.8477	0.8641
19.	0.8468	0.8983	36.	0.8049	0.8384
20.	0.8571	0.8663	37.	0.8122	0.8534
21.	0.7719	0.8116	38.	0.8000	0.8657
22.	0.7647	0.8235	39.	0.7792	0.8040
23.	0.6923	0.7506	40.	0.8571	0.8604
24.	0.7907	0.8127	41.	0.8000	0.8487
25.	0.7925	0.8607			

$$\bar{\rho}(X, Y) = 0.787$$

$$\bar{\rho}(X, Z) = 0.823$$

$$\bar{\rho}(X, Y) = 0.825$$

$$\bar{\rho}(X, Z) = 0.851$$

TABELA 3. Współczynniki podobieństw ρ dla odpowiedzi na 15 lokalnych kwerend po modyfikacji opisów w rozproszonej przestrzeni

Podsystem S_1			Podsystem S_2		
nr kwerendy	$\rho(X, Y)$	$\rho(X, Z)$	nr kwerendy	$\rho(X, Y)$	$\rho(X, Z)$
1.	0.9057	0.9318	1.	0.9254	0.9621
2.	0.8182	0.8182	2.	0.9000	0.8974
3.	0.9677	0.9683	3.	0.9323	0.9353
4.	0.9231	0.9966	4.	0.9231	0.9441
5.	0.9159	0.9540	5.	0.9362	0.9767
6.	0.9000	0.9212	6.	0.9595	0.9545
7.	0.8889	0.8889	7.	0.9120	0.9072
8.	0.9480	0.9690	8.	0.9231	0.9672
9.	0.9100	0.9422	9.	0.9412	0.9197
10.	0.7324	0.7576	10.	0.9481	0.9537
11.	0.9000	0.9466	11.	0.9383	0.9663
12.	0.7742	0.8020	12.	0.8665	0.8796
14.	0.9139	0.9232	13.	0.9412	0.9502
15.	0.9444	0.9678	14.	1.0000	1.0000
			15.	0.9333	0.9533

$$\bar{\rho}(X, Y) = 0.9 \quad \bar{\rho}(X, Z) = 0.923 \quad \bar{\rho}(X, Y) = 0.924 \quad \bar{\rho}(X, Z) = 0.933$$

TABELA 4. Współczynniki podobieństw ρ dla list inwersyjnych po modyfikacji opisów w rozproszonej przestrzeni

Podsystem S_1			Podsystem S_2		
nr listy	$\rho(X, Y)$	$\rho(X, Z)$	nr listy	$\rho(X, Y)$	$\rho(X, Z)$
1.	1.0000	1.0000	1.	1.0000	1.0000
2.	0.8333	0.8333	19.	0.8468	0.8913
3.	0.7711	0.7861	20.	0.8571	0.8737
4.	0.8182	0.8388	21.	0.7719	0.8276
5.	0.7887	0.7887	22.	0.7647	0.8648
6.	0.7536	0.7676	23.	0.6923	0.7201
7.	0.8750	0.8750	24.	0.7907	0.8135
8.	0.8376	0.8427	25.	0.7925	0.8223
9.	0.7500	0.7748	26.	0.8515	0.8770
10.	0.7333	0.7557	27.	0.8421	0.8421
11.	0.6667	0.6928	28.	0.8571	0.8828
12.	0.7500	0.7775	29.	0.8000	0.7944
13.	0.8000	0.8135	30.	0.8250	0.8292
14.	0.5714	0.5714	31.	0.8649	0.8712
15.	0.7273	0.7241	32.	0.8197	0.8127
16.	0.7862	0.8011	33.	0.8615	0.8735
17.	0.8125	0.8543	34.	0.8108	0.8228
18.	0.8267	0.8623	35.	0.8477	0.8497
19.	0.8468	0.8913	36.	0.8049	0.8414
20.	0.8571	0.8737	37.	0.8122	0.8511
21.	0.7719	0.8276	38.	0.8000	0.8621
22.	0.7647	0.8648	39.	0.7792	0.7904
23.	0.6923	0.7201	40.	0.8571	0.8554
24.	0.7907	0.8135	41.	0.8000	0.8254
25.	0.7925	0.8223			

$$\bar{\rho}(X, Y) = 0.792 \quad \bar{\rho}(X, Z) = 0.802 \quad \bar{\rho}(X, Y) = 0.811 \quad \bar{\rho}(X, Z) = 0.830$$

MODELS AND MODIFICATIONS OF DISTRIBUTED INFORMATION RETRIEVAL SYSTEMS BASED ON THESAURI WITH WEIGHTS

The problem of combining n local information systems $S_1, S_2, \dots, S_n$ based on thesauri with weights in one system S is considered. The distributed information retrieval system S may be assumed to be a network of the local systems S_j . Both the generalization relation on the set of descriptor and weights of descriptor in descriptions of documents are taken into account. The fundamental properties of distributed system are described.

The inverted file structure is often used to organize data in the information retrieval system. Operations on inverted lists are modified in order to use them in the distributed information system. The presented retrieval rules may be viewed as the logical approach in implementing a physical distributed information retrieval system. Geometrical interpretation of distributed system is given.

Modification of descriptors weights in document description according to subjective user opinion allows us to prepare new document description, which is more relevant to its contents. Methods of document description modifications are described. Modified document space is also beneficial to "new" users who gain from "knowledge" acquired by system during its past work. Finally, a method of reducing document redundancy and costs of queries in the distributed system are given.

МОДЕЛИ И МОДИФИКАЦИИ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННО-ПОИСКОВЫХ СИСТЕМ. ОСНОВАННЫХ НА ТЕЗАУРАХ С ВЕСАМИ

Существующие информационно-поисковые системы создались, как правило, независимо друг от друга. Соединение многих независимых информационно-поисковых систем в одну большую распределенную систему, доступную в вычислительной сети, требует решения многих проблем не возникающих в традиционных, централизованных информационно-поисковых системах.

В монографии анализируются механизмы поиска ответа в распределенных системах. Рассматривается проблема объединения n локальных информационно-поисковых систем, основанных на тезаурусах с весами, строится распределенная информационно-поисковая система и определяется логическая структура механизма получения глобального ответа в распределенной системе на основании ответов из локальных подсистем. Проводится модификация классического метода инвертированных списков, учитывающая отношение обобщения и веса дескрипторов в описаниях документов. Обозреваются различные распределенные системы.

Применяется динамическое пространство документов, в котором документы могут менять свое местоположение, согласно требованиям пользователей. Дается геометрическая интерпретация распределенной системы. Обновление весов дескрипторов в описании документов на основании субъективной оценки, удачности ответа сделанной пользователем, дает возможность создавать описания лучше отражающие содержание документов.

Обсуждаются способы модификации описаний документов на основании совершаемых пользователями оценок релевантности документов принадлежащих ответу. Из модифицированного пространства документов извлекают выгоду не только пользователи, которые принимали участие в модифицировании документов. "Опыт", накопленным во время предыдущих поисков, оказывается полезным также для новых пользователей. Приводится также статистический критерий уменьшения избыточности в поисковых структурах и оценивается пессимистическая стоимость запросов в распределенной системе.

SPIS TREŚCI

WSTĘP	4
Przedmiot pracy	4
Cel i wyniki pracy	8
1. PRZEGLĄD ZAGADNIEN W TEORII WYSZUKIWANIA INFORMACJI	10
1.1. Klasyfikacja i indeksowanie dokumentów	10
1.2. Modele systemów wyszukiwania informacji	11
2. MATEMATYCZNY MODEL PODSYSTEMU WYSZUKIWANIA INFORMACJI	21
2.1. Opis lokalnego systemu wyszukiwania informacji	21
2.2. Podobieństwo kwerend i opisów dokumentów	24
2.3. Własności odpowiedzi w lokalnych podsystemach	28
2.4. Organizacja lokalnych list inwersyjnych	29
3. WŁASNOŚCI ROZPROSZONEGO SYSTEMU WYSZUKIWANIA INFORMACJI OPARTEGO NA TEZAURUSIE Z WAGAMI	40
4. ORGANIZACJA LIST INWERSYJNYCH W ROZPROSZONYM SYSTEMIE WYSZUKIWANIA INFORMACJI	49
5. HETEROGENICZNE MODELE SYSTEMÓW WYSZUKIWANIA INFORMACJI	54
5.1. Rozproszony system wyszukiwania informacji oparty na tezarusach	54
5.2. Rozproszony system wyszukiwania informacji oparty na modelach wagowych	57
5.3. Rozproszony system wyszukiwania informacji oparty na zbiorach terminów	60
6. ROZPROSZONY SYSTEM WYSZUKIWANIA INFORMACJI OPARTY NA HETEROGENICZNYCH PODSYSTEMACH	63
6.1. Połączenie typu S-ST	64
6.2. Połączenie typu S-SH	66

6.3. Połączenie typu S-SH	67
6.4. Połączenie typu S-SHH	69
6.5. Przykład rozproszonego systemu opartego na heterogenicznych podsystemach	70
7. GEOMETRYCZNY MODEL ROZPROSZONEGO SYSTEMU WYSZUKIWANIA INFORMACJI	73
7.1. Oznaczenia i definicje	73
7.2. Konstrukcja globalnej przestrzeni dokumentów	75
7.3. Własności deskryptorowej przestrzeni dokumentów	76
7.4. Lokalne podprzestrzenie opisów dokumentów	78
8. MODYFIKACJA PRZESTRZENI DOKUMENTÓW	81
8.1. Metody modyfikacji przestrzeni dokumentów	81
8.2. Symulacja procesu wyszukiwania informacji w rozproszonym systemie	86
8.3. Analiza wyników symulacji systemu	87
8.3.1. Modyfikacja lokalnych podprzestrzeni dokumentów a efektywność rozproszonego systemu	89
8.3.2. Modyfikacja rozproszonej przestrzeni dokumentów a efektywność lokalnych podsystemów	91
8.4. Uwagi	92
9. REDUNDANCJA W STRUKTURACH WYSZUKIWAWCZYCH	94
9.1. Redundancja a efektywność systemu	94
9.2. Algorytm wyboru wartości progowych	96
10. KOSZTY KWEREND	100
11. ZAKOŃCZENIE	106
LITERATURA	108
DODATEK	114

CONTENTS

INTRODUCTION	4
Subject of the work	4
Goals and results	8
1. AN OVERVIEW OF INFORMATION RETRIEVAL SUBJECTS	10
1.1. Classification and indexing of documents	10
1.2. Models of information retrieval systems	11
2. MATHEMATICAL MODEL FOR INFORMATION RETRIEVAL SUBSYSTEMS	21
2.1. Description of local information retrieval system	21
2.2. Similarity of queries and document description	24
2.3. Properties of responses in local subsystems	28
2.4. Organization of local inverted files	29
3. PROPERTIES OF DISTRIBUTED INFORMATION RETRIEVAL SYSTEM BASED ON THESAURI WITH WEIGHTS	40
4. ORGANIZATION OF INVERTED FILES IN A DISTRIBUTED INFORMATION RETRIEVAL SYSTEM	49
5. HETEROGENEOUS MODELS FOR THE INFORMATION RETRIEVAL SYSTEMS	54
5.1. Distributed information retrieval system based on thesauri	54
5.2. Distributed information retrieval system based on weights models	57
5.3. Distributed information retrieval system based on terms sets	60
6. DISTRIBUTED INFORMATION RETRIEVAL SYSTEM BASED ON HETEROGENEOUS SUBSYSTEMS	63
6.1. S-ST connection	64
6.2. S-SH connection	66

6.3. S-SH connection	67
6.4. S-SHH connection	69
6.5. Example of distributed system based on heterogeneous subsystem	70
7. GEOMETRICAL MODEL OF DISTRIBUTED INFORMATION RETRIEVAL SYSTEM . .	73
7.1. Notation and basic definition	73
7.2. Construction of global document space	75
7.3. Properties of descriptor space of document	76
7.4. Local space of document descriptions	78
8. MODIFICATION OF DOCUMENT SPACE	81
8.1. Modifications of document space methods	81
8.2. Simulation of information retrieval process in distributed system	86
8.3. A result analysis of distributed system's adaptation	87
8.3.1. Modifications of local document space and effectiveness of distributed system	89
8.3.2. Modification of distributed document space and effectiveness of local subsystems	91
8.4. Remarks.	92
9. REDUNDANCY IN RETRIEVAL STRUCTURES	94
9.1. Redundancy and effectiveness of system	94
9.2. Algorithm for choice threshold values	96
10. COSTS OF QUERIES	100
11. CONCLUSION	106
BIBLIOGRAPHY	108
APPENDIX	114

PRACE NAUKOWE CENTRUM OBLICZENIOWEGO
(wydane w latach 1980—1989)

Nr 1, Studia i Materiały nr 1, <i>Zagadnienia oprogramowania systemów komputerowego wspomagania dydaktyki</i> , Wrocław 1980	20,—
Nr 2, Studia i Materiały nr 2, <i>Zagadnienia oprogramowania węzła pod-sieci komunikacyjnej sieci komputerowej MSK</i> , Wrocław 1983	180,—
Nr 3, Konferencje nr 1, <i>Sieci komputerowe. Usługi, protokoły, modele. Część 1</i> , Wrocław 1985	160,—
Nr 4, Konferencje nr 2, <i>Sieci komputerowe. Usługi, protokoły, modele. Część 2</i> , Wrocław 1986	250,—
Nr 5, Konferencje nr 3, <i>Sieci komputerowe. Teoria, technika, zastosowania</i> , Wrocław 1987	335,—
Nr 8, Monografie nr 1, Z. Huzar, <i>Programowanie procesów komunikujących się w czasie rzeczywistym</i> , Wrocław 1989	300,—
Nr 9, Konferencje nr 4, <i>Sieci komputerowe. Projektowanie, zastosowania, eksploatacja</i> , Wrocław 1989	1000,—

Subscription should be sent (at any time of the year) to:
„Ars Polona”

Krakowskie Przedmieście 7, 00-068 Warszawa

Bank account number: PEK XIII Oddz. W-wa 370044-1195-139-11

Zamówienia na prenumeratę można składać:

OR PAN, PKiN, 00-901 Warszawa

Nr konta bankowego: PEK IX Oddz. W-wa, 370031-4792

Wydawnictwa Politechniki Wrocławskiej
ma stale na składzie Księgarnia Wr 49
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
oraz Wojewódzka Księgarnia Techniczna
ul. Świdnicka 3, 50-067 Wrocław

ISSN 0860-7311