Politechnika Wrocławska

Wydział Informatyki i Zarządzania

kierunek studiów: Informatyka
specjalność: Systemy informacyjne

Praca dyplomowa - magisterska

Problemy rekomendacyjno-wyszukiwawcze w systemach do zarządzania informacją osobistą

Krzysztof Matuszek

słowa kluczowe:
informacja relewantna, rekomendacja, wyszukiwanie
klasyfikacja, model przestrzeni wektorowej
sprzężenie zwrotne, ukryta analiza semantyczna

krótkie streszczenie:
Niniejsza praca poświęcona jest problemom wyszukiwania informacji w systemach do zarządzania informacją osobistą. W pracy przedstawiono ogólny zarys i porównanie współczesnych podejść do zagadnień wyszukiwania i rekomendacji informacji. Część badawcza pracy skupia się na porównaniu 16 strategii wyszukiwania informacji z użyciem modelu przestrzeni wektorowej.

opiekun pracy dyplomowej	..... ***dr inż. Andrzej Gawrych-Żukowski***	..... *ocena*	..... *podpis*

*Do celów archiwalnych pracę dyplomową zakwalifikowano do:**
a) kategorii A (akta wieczyste)
b) kategorii BE 50 (po 50 latach podlegające ekspertyzie)
** niepotrzebne skreślić*

pieczątka Instytutu, w którym student wykonywał pracę

Wrocław 2014

Spis treści

Streszczenie

Niniejsza praca poświęcona jest problemom wyszukiwania informacji w systemach do zarządzania informacją osobistą. W części teoretycznej pracy dokonano klasyfikacji systemu do zarządzania informacją osobistą oraz przedstawiono ogólny zarys i porównanie współczesnych podejść do zagadnień wyszukiwania informacji. W skrócie zaprezentowano szeroką gamę technik wyszukiwawczych oraz opisano typowe problemy informacyjne, wpływające na skuteczność wyszukiwania informacji. Następnie do celów badawczych zdefiniowano możliwe kryteria oceny algorytmów wyszukiwania informacji. Celem części badawczej było utworzenie reprezentacji wektorowej dokumentów tekstowych oraz szczegółowe porównanie różnych strategii wyszukiwania informacji z użyciem modelu przestrzeni wektorowej. Do podstawowego porównywania dokumentów tekstowych skorzystano z kosinusoidalnej miary podobieństwa. W badaniach zaimplementowano oraz skorzystano z takich rozwiązań jak technika Tf-idf, słowniki morfologiczne, algorytmy stemmingu, sprzężenie zwrotne, tworzenie profilów metodą Rocchio oraz ukryta analiza semantyczna. Końcowym rezultatem badań jest porównanie powyższych strategii z użyciem F-miary jako wspólnego kryterium oceny.

Abstract

This document is dedicated to information retrieval problems in personal information management systems. In theoretical basis it gives an introduction to classification of personal information management system and provides a brief comparison of the existing approaches and techniques in information retrieval area. It specifically explains the fundamental information problems influencing the accuracy of information discovery. For research purpose, the basic criteria were introduced for evaluation of information retrieval algorithms. The main scope of research was to create vector representation of text document and evaluate different information retrieval strategies with usage of vector space model. Cosine similarity was basic measure used to evaluate the similarity of text documents. The common techniques like Tf-idf, morphological dictionaries, stemming algorithms, relevance feedback, Rocchio profile creation method and latent semantic analysis were implemented and used for research purpose. The final research aimed to provide a detailed comparison of different retrieval strategies with F-Measure as a main criteria.

Wstęp

Obecnie pod definicją systemów PIM (ang. Personal Information Management – Zarządzanie Informacją Osobistą), kryje się określenie zbioru systemów posiadających podstawowe funkcjonalności takie jak zarządzanie kontaktami, wydarzeniami, spotkaniami, zadaniami czy notatkami. Systemy te umożliwiają użytkownikowi dostęp do takich operacji jak odpowiedni zapis, odczyt oraz możliwość bezpiecznego przechowania wszelkiej informacji, którą użytkownik chce zarządzać.

Użytkownik jest codziennie zalewany ogromną ilością informacji, dlatego decydując się na korzystanie z systemu do zarządzania informacją osobistą, powinien mieć możliwość w bardzo krótkim czasie wyłuskania tych informacji, które go w danym momencie rzeczywiście interesują. Kryteria przeszukiwania powinny być elastyczne oraz rezultat zapytania powinien być wyświetlony w jak najbardziej dogodnej formie. Głównym problemem dla systemów wyszukiwawczych jest umożliwienie użytkownikowi jak najdokładniejszego zdefiniowania swojej potrzeby informacyjnej, aby rezultat wyszukiwania był najbardziej zbliżony do tego, co użytkownik chciał osiągnąć.

Bardzo istotny jest fakt, że systemy do zarządzania informacją osobistą muszą najczęściej indeksować dane częściowo strukturyzowane, czyli w głównej mierze dane tekstowe. Z tego powodu niezbędne jest wypracowanie technik, które pozwolą na odpowiedzenie na potrzebę informacyjną użytkownika właśnie dla tego typu danych. Informację, którą poszukuje użytkownik, można zdefiniować jako informację relewantną. Zatem głównym problemem systemów wyszukiwawczych operujących na danych tekstowych jest zdefiniowanie funkcji, która pozwoli w sposób jak najbardziej zbliżony do potrzeb użytkownika, na określenie relewantności danego dokumentu tekstowego dla danego zapytania.

Aby skutecznie operować na danych niestrukturyzowanych i odpowiadać na tego rodzaju powyższe potrzeby, należy sięgnąć w tym celu do osiągnięć z dziedziny wyszukiwania informacji (ang. information retrieval), operującej w głównej mierze na niestrukturyzowanych danych tekstowych. W niniejszej pracy w głównej mierze zostaną zaprezentowane techniki wyszukiwania informacji opierające się na modelu przestrzeni wektorowej.

Cel i zakres pracy

Głównym celem pracy będzie porównanie różnych strategii wyszukiwania informacji z użyciem modelu przestrzeni wektorowej.

Aby cel został w pełni zrealizowany, wyodrębniono następujące zadania:

- Omówienie podstawowych aspektów związanych z informacją, kategoryzacją informacji oraz systemami do zarządzania informacją osobistą
- Zdefiniowanie podstawowych problemów informacyjnych w zarządzaniu informacją osobistą
- Dokonanie prezentacji podstawowych pojęć i klasyfikacji związanej z zagadnieniami wyszukiwania i rekomendacji informacji
- Opracowanie przeglądu wybranych technik optymalizacji wyszukiwania informacji
- Implementacja wybranych metod
- Zdefiniowanie i implementacja spójnego kryterium oceny strategii wyszukiwania informacji
- Zebranie przykładowych dokumentów testowych oraz przetworzenie ich do reprezentacji wektorowej
- Przeprowadzenie serii eksperymentów dla różnych konfiguracji strategii wyszukiwania informacji oraz ocena wyników
- Porównanie wyników różnych strategii oraz wybranie najlepszej metody wyszukiwania informacji na podstawie ustalonego kryterium oceny

Porównane zostaną następujące sposoby wyszukiwania informacji:

- Wyszukiwanie z użyciem kosinusoidalnej miary podobieństwa
- Wyszukiwanie z użyciem najczęściej występujących termów

Dodatkowo, w ramach serii eksperymentów zostaną porównane następujące czynniki:

- Normalizacja dokumentów tekstowych z użyciem algorytmów morfologicznych oraz algorytmów stemmingu
- Zastosowanie techniki Tf-idf
- Wykorzystanie sprzężenia zwrotnego
- Wykorzystanie techniki ukrytej analizy semantycznej
- Korzystanie z profilów kategorii

Struktura pracy

Rozdział pierwszy omawia szczegółowo definicję informacji, kluczowe czynności związane z informacją oraz typowe problemy informacyjne. Zdefiniowano również pojęcie relewantności informacji oraz danych niestrukturyzowanych, które będą miały kluczowe znaczenie w dalszej części pracy.

Kolejny z rozdziałów przedstawia systemy do zarządzania informacją osobistą jako przykład systemów informacyjno-wyszukiwawczych, opisując szczegółowo, dlaczego powyższe systemy mogą być wykorzystywane do wyszukiwania i rekomendowania informacji relewantnych dla użytkownika. W rozdziale tym zaprezentowano również dwa przykładowe systemy do zarządzania informacją osobistą. Omówiono także podstawowe typy informacji w systemach do zarządzania informacją osobistą oraz wymieniono rodzaje danych i metadanych charakterystyczne dla podstawowych typów informacji

Rozdział trzeci skupia się na próbie zdefiniowania podstawowych sposobów kategoryzacji informacji, jakimi powinien charakteryzować się system do zarządzania informacją osobistą. W pracy zaproponowano następującą systematykę sposobów kategoryzacji informacji: komentarz, tagowanie, struktura drzewiasta, relacje pomiędzy informacjami, asocjacja oraz priorytetowanie.

Rozdział czwarty omawia podstawowe problemy informacyjne w kontekście systemów do zarządzania informacją osobistą, wpływające negatywnie na skuteczność wyszukiwania i rekomendowania informacji relewantnych dla użytkownika. Szczegółowo omówiono problemy informacji odseparowanych oraz informacji nieprawidłowych i niezweryfikowanych.

Rozdział piąty opisuje podstawową klasyfikację rekomendacji oraz opisuje rekomendacje w kontekście systemów do zarządzania informacją osobistą. Szczegółowo opisuje podobieństwa oraz różnice pomiędzy sposobami rekomendacji w serwisach społecznościowych oraz w systemach do zarządzania informacją osobistą. W rozdziale tym odniesiono się do technik rekomendacji wykorzystujących metody kolaboratywne oraz metody oparte na treści.

Rozdział szósty prezentuje podstawowe techniki związane z zagadnieniami wyszukiwania i rekomendacji informacji. Omówiono szczegółowo sposób tworzenia reprezentacji wektorowej dokumentu oraz zaprezentowano różnice wynikające z normalizacji dokumentów tekstowych z użyciem algorytmu stemmingu Snowball oraz słownika morfologicznego Ispell. W ramach przeglądu zaprezentowano kilka modeli wyszukiwania informacji. Szczegółowej analizie zostały poddane model boolowski oraz model przestrzeni wektorowej. W ramach zagadnienia optymalizacji wyników wyszukiwania, opisano sposób użycia słowników oraz technikę Tf-idf, pozwalającą na odpowiednie zbilansowanie wag termów na podstawie ich wartości dyskryminacyjnej.

Rozdział siódmy opisuje podstawowe sposoby testowania skuteczności algorytmów wyszukiwania informacji, testowanie AB oraz testowanie asercji relewancji. W rozdziale tym opisano szczegółowo metodykę Cranfield, bazującą na testowaniu asercji relewancji, umożliwiającą przeprowadzenie w pełni powtarzalnych serii eksperymentów, pozwalających na porównywanie różnych strategii wyszukiwania informacji. W rozdziale tym omówiono również podstawowe funkcje oceny wyszukiwania informacji. Szczegółowo opisano funkcje precyzji, zwrotu, skuteczności, F-miary oraz R-precyzji.

Rozdział ósmy w skrócie prezentuje systematykę klasyfikacji dokumentów tekstowych oraz prezentuje sposób klasyfikacji dokumentów z użyciem profilów metodą Rocchio. W rozdziale tym zaprezentowano również podstawowy podział sposobów tworzenia profilów klas.

Rozdział dziewiąty opisuje szczegółowo technikę sprzężenia zwrotnego oraz prezentuje algorytm Rocchio, pozwalający na modyfikację wstępnego zapytania. W rozdziale tym zaprezentowano również trzy najbardziej skuteczne strategie wyszukiwania informacji, które opublikowała Eleanor Ide na bazie algorytmu Rocchio [21].

Rozdział dziesiąty poświęcony jest zagadnieniu ukrytej analizy semantycznej, wraz z dokładnym opisem procesu tworzenia zredukowanej macierzy term-dokument. Zaprezentowano również geometryczną reprezentację przykładowych dokumentów oraz zapytań przy użyciu zredukowanej macierzy term-dokument.

W rozdziale jedenastym przedstawiono kolejne etapy związane z przeprowadzonymi badaniami. Na początku został przedstawiony dokładny cel badań wraz z opisem kolejnych etapów dla każdej badanej strategii. Następnie omówiono zebrane dane testowe, wyznaczono wspólne kryterium oceny do porównywania strategii wyszukiwania informacji oraz opisano podstawowe założenia przyjęte w trakcie badań. Dalsza część rozdziału prezentuje dokładny opis metodologii badań, wyniki oraz ich omówienie.

Rozdział dwunasty podsumowuje całość badań. W rozdziale tym zawarto ocenę zaprezentowanych podejść oraz wskazano dalsze kierunki badań. Zakończeniem pracy jest wykres podsumowujący, w którym zaprezentowano porównanie każdej strategii z uzyskaną najlepszą możliwą dla niej konfiguracją, pozwalającą na uzyskanie najwyższej wartości średniej F-miary.

1. Informacja

Celem tego rozdziału jest wprowadzenie podstawowych definicji i zaprezentowanie klasyfikacji związanych z informacją, występujących w literaturze. Zapoznanie się z takim wprowadzeniem teoretycznym ułatwi dalsze zrozumienie problemów związanych z przetwarzaniem informacji oraz zrozumienie pojęć w dalszej części pracy.

1.1. Opis informacji

Podstawowym tematem pracy jest informacja, która zostanie omówiona szczegółowo w tym podrozdziale.

Informacja to dane zawarte w komunikacie, zinterpretowane przez odbiorcę, mające dla niego znaczenie i wnoszące do jego świadomości element nowości, czyli zmniejszające jego niewiedzę. By dane stały się informacją niezbędny jest ich odbiorca, który decyduje, po pierwsze czy chce dane zinterpretować, po drugie czy są one dla niego zrozumiałe i w jakim stopniu. Wtedy dane stają się dla odbiorcy wiadomością. Następnie odbiorca określa, czy wiadomość jest powtórzeniem czegoś co już wie, czy też stanowi dla niego element nowości. W przypadku twierdzącym, wiadomość staje się informacją. Ponieważ informacja zależy od zdolności interpretacyjnych odbiorcy, ma ona charakter subiektywny. [1]

Każda informacja posiada podstawowe własności: [1]

- informacja jest pojęciem niematerialnym, a jej ujawnienie wymaga danych
- istnieje niezależnie od subiektywnego odbioru, a więc jest obiektywna
- ma różne znaczenie dla różnych odbiorców
- jest odzwierciedleniem pewnej cechy obiektu lub wycinkowym jego opisem
- stanowi uproszczony model określonego wycinka rzeczywistości
- przejawia cechę synergii
- jest łączona przez odbiorców ze znanymi innymi informacjami
- mimo że nie jest energią, podlega ilościowemu pomiarowi
- wykazuje się różnorodnością, wynikającą z odmienności rozpatrywanych obiektów, rozmaitości źródeł, subiektywizmu interpretacyjnego odbiorców
- podlega powielaniu i przenoszeniu w czasie i przestrzeni, a także przetwarzaniu, nie ulegając zniszczeniu, lecz najwyżej zniekształceniom i deformacjom
- jest zasobem niewyczerpalnym, co wynika z możliwości jej powielania, nieskończonej liczby obiektów i ich nieskończonej złożoności
- pozyskiwanie, przechowywanie, przesyłanie i udostępnianie informacji wymaga określonych kosztów
- rozkład informacji w otoczeniu jest nierównomierny, co wywołuje jej asymetrię, czyli niejednakową dostępność dla różnych odbiorców, ze względu na źródła i koszty jej pozyskania, preferencje w ustalaniu faktów i inne czynniki

Informacja jest pragmatyczna, jeżeli jednoznacznie odwzorowuje rzeczywistość (relacja informacja – obiekt rzeczywisty), zaspokaja potrzeby użytkownika (relacja informacja – wiedza) i umożliwia podejmowanie efektywnych działań. [2]

Pojęcie informacji jest bardzo szerokie, dlatego mając świadomość, że powyższe opisy nie oddają w pełni definicji informacji, w tab. 1.1 zaprezentowano wybrane dodatkowe definicje informacji:

Definicja	Źródło	Kontekst
„Wiadomość, wieść, nowina, rzecz zakomunikowana, zawiadomienie, komunikat; pouczenie, powiadomienie, zakomunikowanie o czymś; dane (...)”	www.slownik-online.pl/kopaliński	Relacja odbicia
„Element wiedzy komunikowany, przekazywany komuś za pomocą języka lub innego kodu; także to, co w danej sytuacji może dostarczać jakiejś wiedzy; wiadomość, komunikat, wskazówka”	*Słownik współczesnego jeżyka polskiego* 1998]	Relacja odbicia
„Informatio (łac.) - wyobrażenie, wyjaśnienie, zawiadomienie - własność przysługująca materialnemu nośnikowi informacji (zwanemu sygnałem), której istotą jest redukcja niepewności; potocznie: konstatacja stanu rzeczy, wiadomość”	*Wielka encyklopedia PWN* 2002|	Relacja odbicia
„Znaczenie (treść), jakie przy zastosowaniu odpowiedniej konwencji przyporządkowuje się danym, tj. liczbom, faktom, pojęciom lub rozkazom ujętym w sposób wygodny do przesyłania, interpretacji lub przetwarzania”	Polska Norma 19711	Relacja odbicia
„Nazwa treści zaczerpnięta ze świata zewnętrznego w procesie naszego dostosowywania się do niego i przystosowywania się do niego naszych zmysłów (...). Informacja nie jest energią ani materią”	Wiener 1961, s. 24	Relacja odbicia
„Nazwa treści doznań zmysłowych i umysłowych człowieka”	Ciborowski 2005, s. 32	Relacja odbicia
„Treść o określonym znaczeniu o czymś, dla kogoś i ze względu na coś, wyrażona za pomocą znaków językowych lub/i pozajęzykowych”	ILyons 1984, s. 60	Relacja odbicia
„Rodzaj zasobów, który pozwala na zwiększenie naszej wiedzy o nas i otaczającym nas świecie”	Kisielnicki i Sroka 2005. s. 131	Relacja realizacji
„Właściwości sygnału lub wiadomości polegające na zmniejszeniu nieokreśloności co do stanu sytuacji lub jego dalszego rozwoju”	Gackowski 1974, s. 37	Relacja realizacji
„Treść zaczerpnięta ze świata zewnętrznego, która zwiększa wiedzę lub zmniejsza niewiedzę decydującego, niepewność i nieokreśloność sytuacji decyzyjnej"	Wierzbicki 1981, s. 9	Relacja realizacji
„Dane o procesach i zjawiskach gospodarczych, wykorzystywane w procesie podejmowania decyzji”	IMesner 1971, s. 10	Relacja realizacji

Tab. 1.1: Wybrane definicje informacji. [3]

1.2. Problemy informacyjne

Informacja jest związana z procesem decyzyjnym. Jest ona tworzywem, z którego powstaje decyzja, ponadto podjęta decyzja jest informacją dla decyzji późniejszych. W procesie decyzyjnym występuje ciąg procesów będących transformacjami informacji, odpowiadającymi fazom procesu decyzyjnego, takim jak: analiza, sformułowanie problemu, rozwiązanie problemu i podejmowanie ostatecznej decyzji. [3]

Aby lepiej zrozumieć proces decyzyjny, z jakim jest związana informacja, należy przytoczyć podstawowe operacje, którym poddawana jest informacja. Poniżej zaprezentowano kluczowe czynności związane z informacją: [4]

- Tworzenie informacji
- Zmiana informacji
- Zapis informacji
- Wyszukiwanie informacji
- Integracja informacji
- Podejmowanie decyzji na podstawie informacji
- Komunikowanie informacji
- Unieważnienie informacji

W procesie tym pojawia się wiele przejawów niesprawności dotyczących informacji jako produktu, jak również procesu. Poniższa systematyka wraz z opisem definicji została opracowana na podstawie [3].

Do podstawowych przejawów niesprawności informacji jako produktu zalicza się:

- przeciążenie informacyjne
- dwuznaczność, wieloznaczność informacji
- anemia informacyjna (osłabione pole widzenia)

Do podstawowych przejawów niesprawności informacji jako procesu informacyjnego zalicza się:

- zaleganie informacji
- dystorsja informacji

Przeciążenie informacyjne występuje wówczas, gdy osoba uzyskuje znacznie więcej informacji niż może wykorzystać. Do negatywnych skutków tego zjawiska zalicza się wydłużenie czasu potrzebnego na wyszukiwanie informacji, zwiększenie kosztów przetwarzania informacji oraz niespójność informacji. Przeciążenie informacyjne odbija się niekorzystnie na jakości podejmowanych decyzji i procesu decyzyjnego.

Dwuznaczność, lub wieloznaczność informacji, występuje w sytuacji możliwości różnej interpretacji tej samej informacji, gdy nie da się ustalić, która z nich jest właściwa. Przyczyną powstawania wieloznaczności informacji jest najczęściej stosowanie różnego rodzaju skrótów, żargonów oraz nieprecyzyjne podawanie danych.

Anemia informacyjna (osłabione pole widzenia) jest przeciwieństwem przeciążenia informacyjnego, oznacza to niedobór informacji. Najłagodniejsza postać anemii informacyjnej przejawia się tym, że informacji jest dużo, lecz są one częściowo zdezaktualizowane. Znacznie bardziej zaawansowaną formą anemii informacyjnej jest zróżnicowanie informacji, ale rozproszonych i niekompletnych, co uniemożliwia ich agregację w uporządkowane zbiory. Z powodu niekompletności informacji, powstaje niepewność informacyjna.

Zaleganie informacji powstaje najczęściej poprzez przeciążenie informacyjne oraz zbyt duży podział procesów informacyjnych. Analiza przeprowadzona w koncernie Siemens w Monachium w połowie lat osiemdziesiątych wykazała, że w tym przedsiębiorstwie 95% czasu trwania procesów informacyjnych to czas zalegania informacji, a tylko 5% to czas związany z ich przetwarzaniem i wykorzystaniem. [5]

Dystorsja informacyjna jest bardziej złożonym przypadkiem wieloznaczności informacji i oznacza świadomie lub nieświadomie dokonaną deformację informacji w trakcie jej przepływu przez kolejne punkty obiegowe. Główną przyczyną jest modyfikacja treści i znaczenia informacji w przekazie informacyjnym.

Główny temat pracy dyplomowej będzie związany z wyszukiwaniem i rekomendacją informacji. Jest to kluczowa czynność związana w przetwarzaniem informacji, która przynosi użytkownikowi realny zysk – dostęp do informacji, których rzeczywiście potrzebuje. Użytkownik powinien mieć możliwość w bardzo krótkim czasie wyłuskania tych informacji, które go w danym momencie interesują. Jak wspomniano wcześniej, każdy z nas jest codziennie zalewany ogromną ilością informacji, co według powyższych definicji, zostało oznaczone terminem przeciążenia informacyjnego, czyli uzyskiwania znacznie więcej informacji niż można wykorzystać.

Przy tak wielkiej ilości nieużytecznych dla użytkownika informacji, problemem staję się dotarcie do najbardziej potrzebnych informacji w celu skorzystania z nich. Celem niniejszej pracy jest próba zredukowania tego zjawiska tak, aby użytkownik uzyskiwał dokładnie tyle informacji ile potrzebuje.

1.3. Informacja w danych

Aby lepiej zrozumieć problemy związane z wyszukiwaniem najbardziej potrzebnych informacji, należy wpierw wyjaśnić definicję danych. Jak opisano wcześniej, informacja to dane zawarte w komunikacie, dlatego w tym podrozdziale zostanie szczegółowo zaprezentowana definicja danych oraz zostaną omówione cechy danych dobrej jakości.

Dane
Dane reprezentują fakty. W systemach zarządzania wspomaganych komputerowo dane są kodowane za pomocą odpowiednich symboli. Mogą być rejestrowane, przetwarzane i przesyłane. Dane są przesyłane do świadomości odbiorcy w postaci komunikatu, zatem każdy komunikat zawiera dane. Choć same dane nie mają znaczenia ani celu, to dobór odpowiednich symboli może narzucać lub sugerować ich określoną interpretację. Dane są wysokiej jakości jeżeli nadają się do użycia zgodnie z przeznaczeniem w zakresie działania, podejmowania decyzji i planowania. Dane nadają się do użycia zgodnie z przeznaczeniem, jeżeli nie zawierają defektów i posiadają pożądane cechy. [6]

Wang i Strong [7] opracowali 15 wymiarów jakości danych z perspektywy użytkownika danych. Podzielono je na cztery kategorie:

Kategoria	Wymiar
Wewnętrzna	dokładność, obiektywność, wiarygodność, reputacja
Dostępu	dostępność, bezpieczeństwo dostępu
Kontekstu	relewancja, wartość dodana, aktualność, kompletność, ilość danych
Reprezentacji	interpretowalność, łatwość zrozumienia, zwięzłość, spójna reprezentacja

Tab. 1.2: Wymiary jakości danych. [7]

- **Dokładność** – zakres w jakim dane są poprawne i odpowiadają rzeczywistości
- **Obiektywność** – zakres w jakim dane są bezstronne i pozbawione tendencyjności
- **Wiarygodność** – zakres w jakim dane postrzegane są jako prawdziwe i poprawne
- **Reputacja** – zakres w jakim dane posiadają wysokie uznanie pod względem źródła lub zawartości
- **Dostępność** – zakres w jakim dane są dostępne lub łatwe do uzyskania
- **Bezpieczeństwo dostępu** – zakres w jakim dostęp do danych został ograniczony, aby zapewnić ich bezpieczeństwo
- **Relewantność** – zakres w jakim informacje zawarte w danych nadają się do stawianych im zadań
- **Wartość dodana** – zakres w jakim wykorzystanie danych przyniesie wymierne korzyści
- **Aktualność** – zakres w jakim dane są aktualne ze względu na potrzeby stawianych im celów
- **Kompletność** – zakres w jakim dane zawierają wszystkie wymagane informacje, zarówno ilościowo jak i jakościowo, potrzebne do stawianych im celów
- **Ilość danych** – zakres w jakim ilość danych wpływa na utrudnienie wykonania operacji na nich
- **Interpretowalność** – zakres w jakim dane są zapisane w odpowiednim języku, przy użyciu odpowiedniej symboliki i z zachowaniem odpowiednich jednostek
- **Łatwość zrozumienia** – zakres w jakim typowy użytkownik jest w stanie zrozumieć informacje zawarte w danych
- **Zwięzłość** – zakres w jakim dane nie zawierają nadmiarowych i zbędnych informacji oraz nie zajmują w sposób nieuzasadniony dużo miejsca
- **Spójna reprezentacja** – zakres w jakim dane przestawiane są w jednolity sposób

Dane o dobrej jakości są podstawowym warunkiem tego, aby informacje, które będzie wyszukiwał użytkownik, były dla niego użyteczne. Jednak z powodu przeciążania informacyjnego, samo istnienie ogromnej ilości danych, choć generalnie o dobrej jakości, nie gwarantuje użytkownikowi wyszukania tych informacji, które rzeczywiście go interesują. Dlatego tak ważnym pojęciem dla wyszukiwania informacji jest relewantność, które jest też jednym z powyższych wymiarów jakości danych i któremu w głównym stopniu zostanie poświęcona dalsza część pracy.

1.4. Relewantność informacji

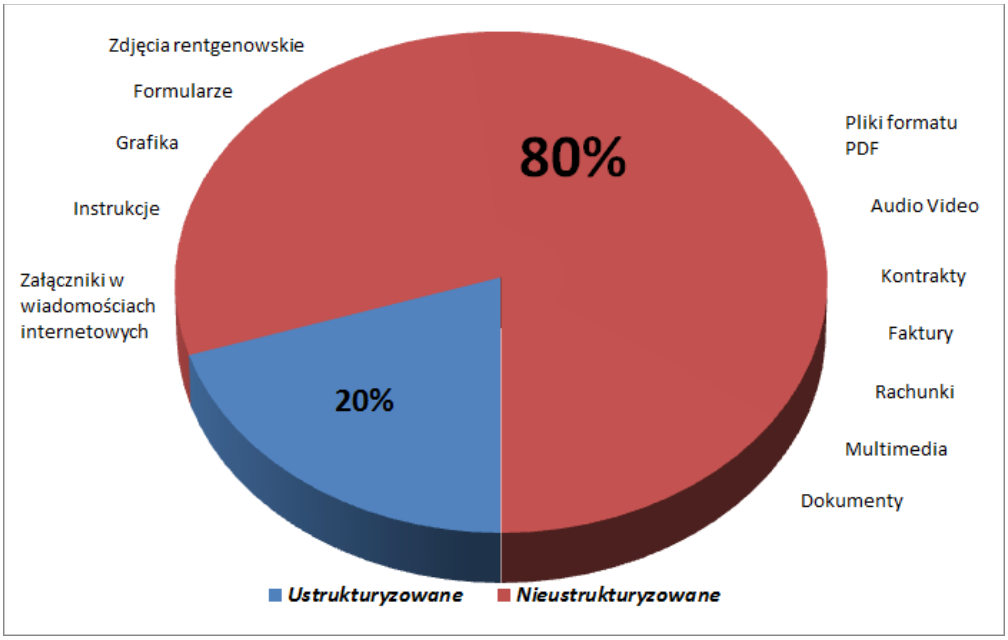
Relewantność informacji jest jednym z najistotniejszych czynników w systemach do zarządzania informacją osobistą. Relewantność informacji jest ważnością informacji w stosunku do danego zapytania. Dana informacja jest relewantna, co oznacza, że spełnia oczekiwania odbiorcy. Wyszukiwarki internetowe nadają stronom internetowym pewien wskaźnik relewancji na podstawie podanego zapytania. Wyniki najbardziej relewantne są umieszczane na samym początku. Podobnie przy zarządzaniu informacją osobistą, użytkownik wyszukując informację, będzie chciał zobaczyć tylko najbardziej relewantne informacje dla danego zapytania, spełniające dane kryterium. Podstawowe cechy relewancji informacji [8]:

- Jej wartość jest subiektywna, zależy od użytkownika
- Może się zmieniać
- Jest zazwyczaj wartością numeryczną

Ponieważ ocena stopnia relewancji jest wartością mierzalną, możliwe jest utworzenie rankingu informacji, posortowanego malejąco według stopnia relewancji. Najbardziej relewantne informacje prezentowane są użytkownikowi na początku rankingu, najmniej relewantne – na końcu. Ułatwia to przeglądanie wyników zapytania złożonych z wielu informacji.

1.5. Strukturyzowane a niestrukturyzowane dane

Dane mogą być strukturyzowane lub niestrukturyzowane, w zależności od tego jak są przechowywane i zarządzane. Dane są niestrukturyzowane, jeżeli nie mogą być zorganizowane w wierszach i kolumnach, zatem nie ma możliwości wykonania zapytania na nich przez aplikacje biznesowe. Dla przykładu, dane o kliencie mogą być przechowywane w wiadomości w poczcie elektronicznej, kartach biznesowych lub nawet elektronicznej formie w dokumentach o rozszerzeniu .docx, .txt czy .pdf. Szacuje się, że około 80 procent danych biznesowych jest niestrukturyzowanych, zatem wymaga to dodatkowych zasobów, aby nimi zarządzać [9].



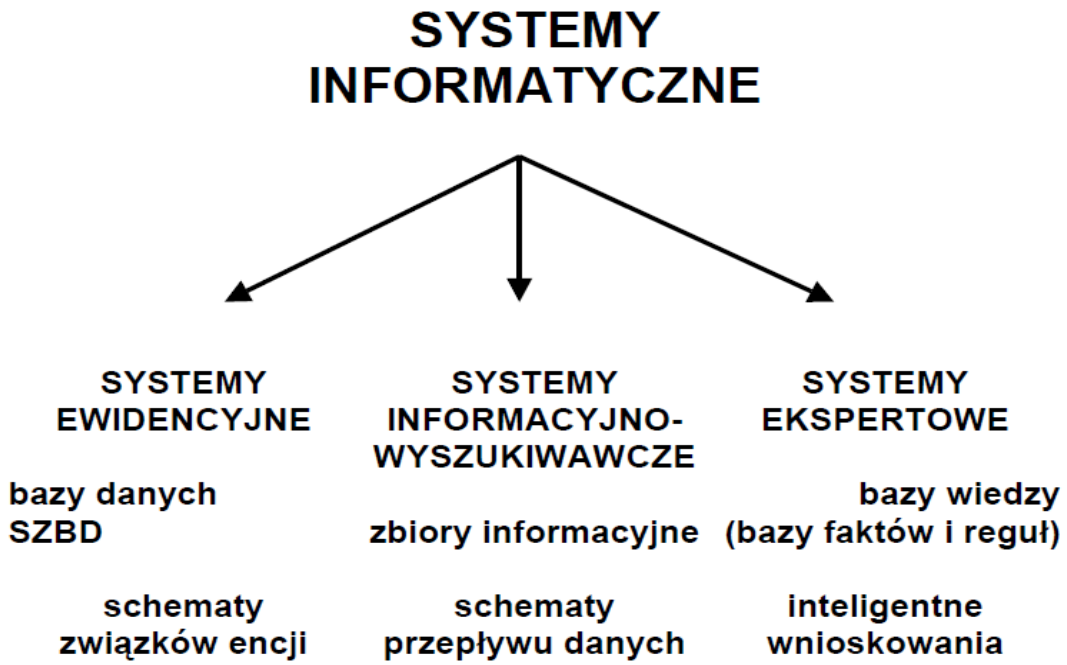
Rys. 1.1: Dane strukturyzowane i niestrukturyzowane. [9]

2. System do zarządzania informacją osobistą

2.1. Klasyfikacja systemu do zarządzania informacją osobistą

System do zarządzania informacją osobistą jest typem oprogramowania, który służy użytkownikowi jako osobisty organizer do efektywnego zarządzania własnymi informacjami, takimi jak notatki, spotkania czy zadania. Bardzo często służy ono jako osobisty kalendarz, pamiętnik, książka adresowa czy program do planowania własnego czasu. Wraz z końcem XX wieku, papierowe organizery zaczęły być zastępowane przez ich cyfrowe odpowiedniki, takie jak PDA (ang. Personal Digital Assistant), elektroniczne menadżery informacji osobistej, czy internetowe systemy do zarządzania informacją osobistą. Proces ten został przyśpieszony w XXI wieku przez gwałtowny wzrost wytwarzanych aplikacji mobilnych, oraz sprzedaży smartphonów i tabletów.

Systemy do zarządzania informacją osobistą należą do grupy systemów informacyjno-wyszukiwawczych (rys. 2.1), charakteryzujących się problemem relewancji pomiędzy różnymi informacjami, ogromną liczbą powiązanych luźno ze sobą informacji oraz bardzo dużą liczbą niewystarczająco precyzyjnych, dostępnych zapytań. Podobnymi do nich systemami, należącymi do tej samej grupy systemów informacyjno-wyszukiwawczych, są katalogi filmów i utworów muzycznych, wyszukiwarki internetowe, katalogi biblioteczne czy systemy matrymonialne. Wyróżniają się one stosunkowo niedużą konsekwencją błędów lub nieaktualności danych. Ich dane są aktualizowane okresowo, często są opcjonalne i powtarzają się, struktura rekordów jest bardzo bogata, zmienia się również liczba i długość pól w rekordzie [10].



Rys. 2.1: Klasyfikacja systemów informatycznych. [10]

W porównaniu do systemów informacyjno-wyszukiwawczych, systemy ewidencyjne charakteryzują się bardzo dużą dyscypliną oraz precyzją z powodu dużej konsekwencji jakiegokolwiek błędu. Ich dane są aktualizowane natychmiastowo, pola są obowiązkowe, ich długość w rekordzie jest stała, również rekordy są stałej długości. Przykładowymi systemami ewidencyjnymi są systemy finansowo-księgowe, systemy sprzedaży, rezerwacji, kontroli ewidencji ludności lub pojazdów [10].

Systemy do zarządzania informacją osobistą, z powodu niedeterministycznego modelu, niewystarczająco precyzyjnych zapytań oraz ogromnej ilości danych, powinny być stosowane głównie w wyszukiwaniu i rekomendowaniu informacji relewantnych dla użytkownika.

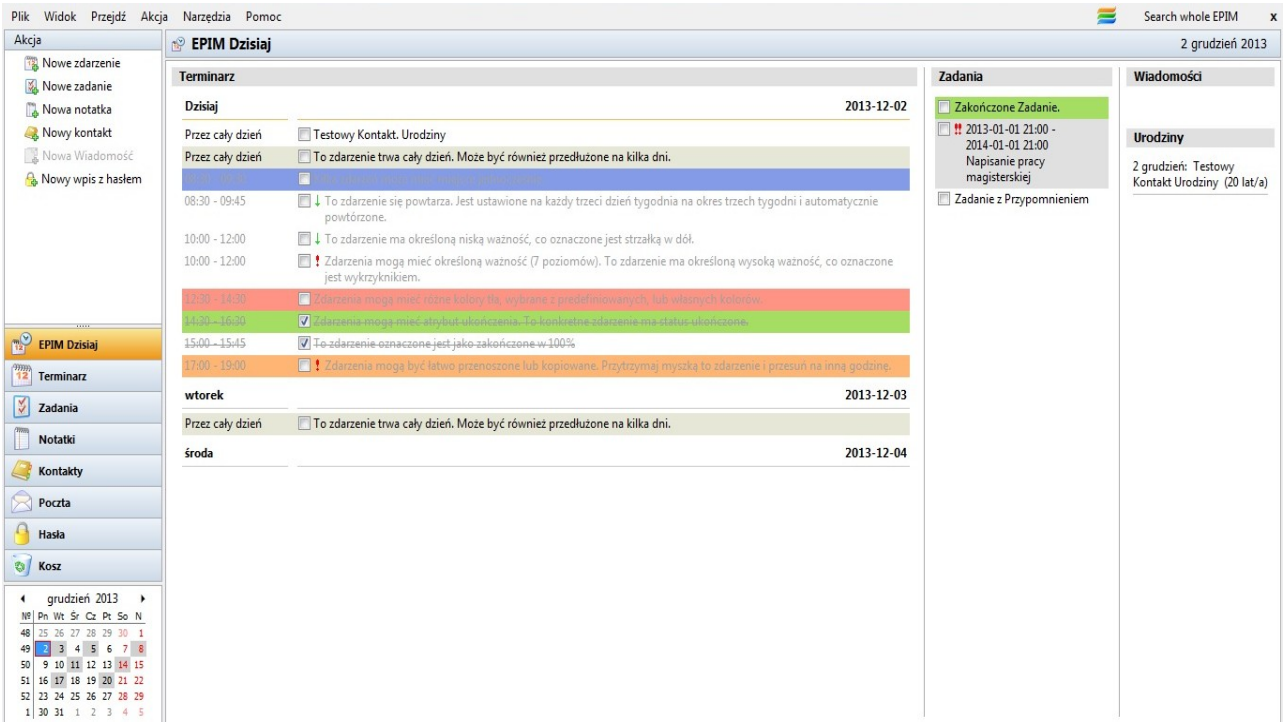
Idealny system do zarządzania informacją osobistą powinien zawsze dostarczać idealnie dopasowane informacje, w idealnej formie przystępnej dla użytkownika, uwzględniając wszystkie wymagane kryteria użytkownika. Zarządzanie informacją osobistą pozwala użytkownikowi zaoszczędzić czas, który w sposób tradycyjny, zostałby poświęcony na czasochłonnych zadaniach takich jak archiwizacja informacji czy ręczne wyszukiwanie informacji. Dzięki systemom do zarządzania informacją osobistą, użytkownik może poświęcić więcej czasu na tworzenie nowych informacji czy wykorzystywanie już istniejących.

2.2. Przykładowe systemy do zarządzania informacją osobistą

W niniejszym podrozdziale zostaną zaprezentowane 2 systemy do zarządzania informacją osobistą. Omówione zostaną następujące systemy:

- EssentialPIM
- Evernote

Powyższe systemy posiadają niestety tylko część z dostępnych sposobów kategoryzacji informacji, opisanych w rozdziale 3. Dostępne jest tworzenie kategorii informacji (tagowanie), tworzenie wielopoziomowej struktury informacji (struktury drzewiastej) oraz tworzenie powiązania (asocjacji) pomiędzy informacjami. Priorytetowanie jest ograniczone tylko do listy zadań.



Rys. 2.2: Zrzut ekranu z systemu EssentialPIM

EssentialPIM

EssentialPIM jest darmowym systemem firmy AstonSoft, pozwalającym na zarządzanie spotkaniami, zadaniami, notatkami, kontaktami, zapisanymi hasłami oraz wiadomościami email, obsługując różne platformy sprzętowe oraz przetwarzanie w chmurze. EssentialPIM został do tej pory wydany na komputer osobisty, Android oraz iPhone/iPAd. Wersja na komputer osobisty posiada wersję instalacyjną oraz wersję przenośną, zabezpieczoną 256 bitowym kluczem AES.

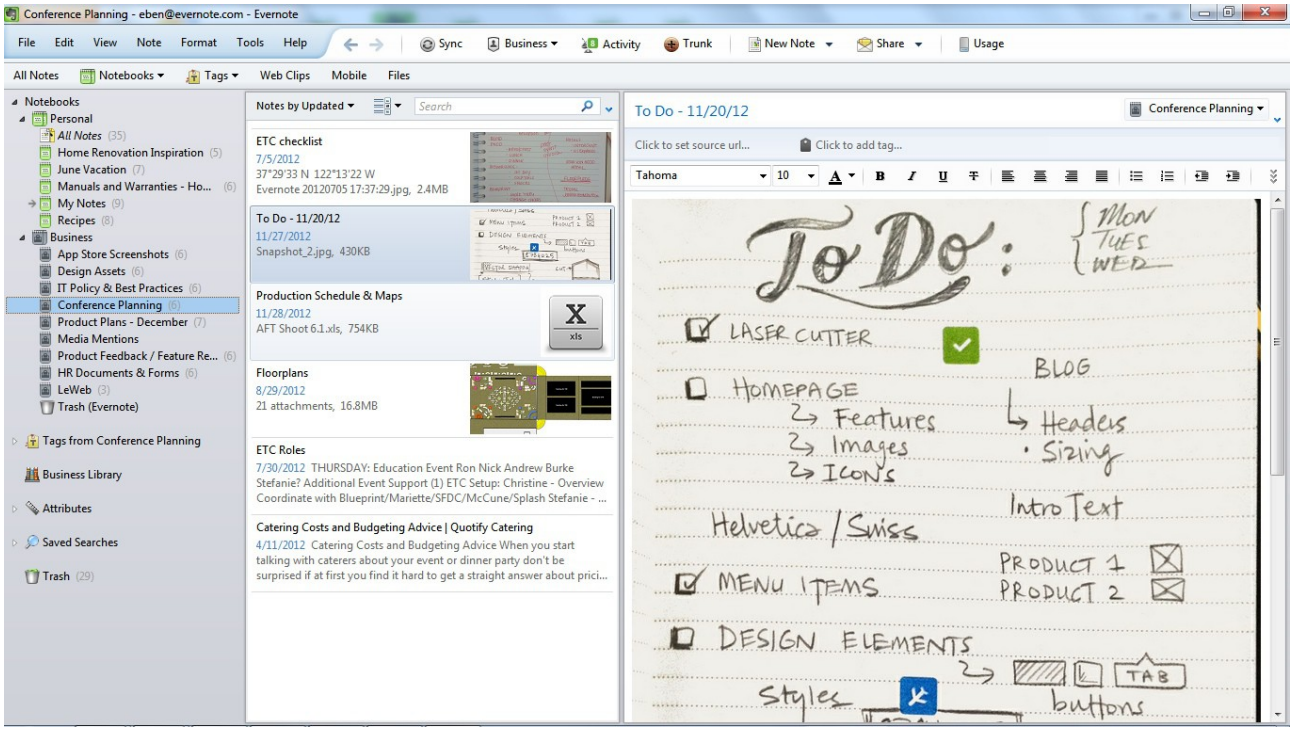
Podstawowa wersja posiada następujące funkcjonalności:

- zarządzanie kalendarzem, zadaniami, notatkami, kontaktami, hasłami, wiadomościami email
- synchronizacja z Android oraz iOS
- możliwość importu i eksportu danych
- listy zadań – priorytetowanie, kategorie, status wykonania
- wielopoziomowa struktura notatek oraz katalogów
- notatki – możliwość wstawiania obrazów, tabel oraz formatowanego tekstu
- kontakty – sortowanie, przeszukiwanie
- szyfrowanie danych
- generowanie haseł

Wersja płatna, komercyjna jest rozszerzona o następujące funkcje:

- przetwarzanie w chmurze
- wieloosobowy dostęp do bazy danych
- zaawansowane kopie bezpieczeństwa i zabezpieczenia szyfrujące
- możliwość utworzenia połączeń pomiędzy dowolnymi danymi
- kategorie – filtrowanie oraz dodatkowe widoki
- listy zadań – filtrowanie, tworzenie hierarchii
- kontakty – możliwość przypisania do dowolnej ilości grup
- grupowanie haseł

EssentialPIM posiada również specjalną wersję sieciową, posiadającą wszystkie funkcje wersji komercyjnej, jednak rozbudowaną o możliwość współpracy wielu użytkowników w sieci. Główną cechą tej wersji jest możliwość współdzielenia swoich danych oraz zaawansowanego zarządzaniami uprawnieniami.



Rys. 2.3: Zrzut ekranu z systemu Evernote

Evernote

Evernote jest darmowym systemem stworzonym przez firmę Evernote, stworzonym głównie do zarządzania notatkami. System Evernote pozwala na synchronizację wszystkich danych pomiędzy wieloma urządzeniami. Notatki, dokumenty, obrazy czy nagrania audio są dostępne dla użytkownika zarówno na komputerach stacjonarnych Windows i Mac, jak i na urządzeniach przenośnych typu iPad, iPhone, Android, Windows Phone i Blackberry. Główną zaletą tego systemu jest możliwość zgromadzenia informacji w jednym miejscu i posiadania do nich swobodnego dostępu z każdego komputera czy urządzenia.

Synchronizacja w systemie Evernote polega na stałej aktualizacji wszystkich komputerów i urządzeń do najnowszej wersji notatki. Kiedy nowa notatka zostaje utworzona lub edytowana na którymkolwiek z urządzeń z zainstalowanym Evernote, zostaje ona przesłana do serwera, skąd wszystkie inne urządzenia pobiorą ją podczas następnej synchronizacji.

Wersja płatna, komercyjna jest rozszerzona o następujące funkcję:

- przechowywanie notatek bez połączenia z internetem
- współdzielenie notek z innymi użytkownikami
- dodatkowa warstwa zabezpieczeń
- zwiększony maksymalny rozmiar notatki
- zwiększony miesięczny limit transferu danych z serwerem
- inteligentne wyszukiwanie tekstu w dokumentach, plikach PDF i obrazach

Evernote posiada również wersję biznesową, przygotowaną specjalnie dla firm. Wersja ta charakteryzuje się centralnym zarządzaniem wiedzą w firmie, co umożliwia poznanie wiedzy członków zespołu jak i wyszukiwanie ekspertów w interesującym dla użytkownika temacie. Wersja biznesowa charakteryzuje się jeszcze bardziej zwiększonym limitem transferu danych jak i możliwością zarządzania kontami wszystkich pracowników w firmie.

2.3. Dane w systemach do zarządzania informacją osobistą

Jak opisano w podrozdziale 1.5, dane mogą być być strukturyzowane lub niestrukturyzowane, z czego większość w systemach do zarządzania informacją osobistą to dane niestrukturyzowane, najczęściej dane tekstowe. Bardzo istotny jest fakt, że systemy to zarządzania informacją osobistą muszą najczęściej indeksować dane częściowo strukturyzowane. Przykładowymi danymi częściowo strukturyzowanymi są dane tekstowe z dodatkowymi opisami np. "tytuł" czy "data powstania". Zapotrzebowanie na takiego rodzaju dane tekstowe w systemach do zarządzania informacją osobistą, różni się znacząco od typowych bazodanowych aplikacji, które operują na bardzo strukturyzowanych danych takich jak płatności czy innego rodzaju dane ewidencyjne. Struktura rekordu w typowo bazodanowych aplikacjach nie tylko determinuje typ przechowywanych danych, lecz także w sposób jasny określa znaczenie danej wartości (np. pole "Nazwisko"). Dla porównania, dane częściowo strukturyzowane, mogą posiadać pewien logiczny podział np. na "tytuł", "opis" i "treść" ale znaczenie "treści" nie jest tak oczywiste jak w danych w typowych systemach ewidencyjnych. Wiadomości e-mail są doskonałym przykładem danych częściowo strukturyzowanych, ponieważ posiadają dobrze zdefiniowane nagłówki (nadawca, odbiorca, tytuł itp.), ale posiadają również niestrukturyzowaną część "treść wiadomości", która jest łańcuchem znaków pozbawionym wewnętrznej struktury.

Każdy typ informacji w systemach do zarządzania informacją osobistą składa się najczęściej z kilku rodzajów strukturyzowanych danych i metadanych oraz danych niestrukturyzowanych. Najprostsza definicja metadanych to dane o danych. Przykładowo, dla bazy danych, metadanymi są definicje tabel, widoków i kluczy. Dla plików przykładową metadaną jest data modyfikacji pliku.

Podstawowe typy informacji w systemach do zarządzania informacją osobistą to:

- notatka
- zadanie
- kontakt
- wiadomość przychodząca
- wiadomość wychodząca
- spotkanie
- plik

Spośród wszystkich typów informacji można wymienić kilka rodzajów danych i metadanych, które są dostępne w każdym z typów informacji, przykładowo:

Rodzaj danych / metadanych
Tytuł
Data powstania
Data ostatniej modyfikacji
Powiązania z inną informacją
Kategorie informacji (tagi)
Identyfikator w systemie
Autor informacji
Uprawnienia do odczytu
Uprawnienia do edycji

Tab. 2.1: Domyślne rodzaje danych i metadanych. Opracowanie własne.

Ponadto każdy z wymienionych wcześniej typów informacji posiada rodzaje danych i metadanych charakterystyczne tylko dla danego typu informacji, przykładowo:

Typy informacji	Rodzaj danych / metadanych
Notatka	Zawartość notatki
	Dodatkowy opis
	Słowa kluczowe
Zadanie	Opis zadania
	Zadania poprzedzające
	Termin do
	Procent wykonania
	Aktualny stan zadania
	Osoby przypisane
	Priorytet
Kontakt	Imię
	Drugie imię
	Nazwisko
	Stanowisko
	Telefony
	Adres zamieszkania
	Rok urodzenia
	Adresy email
	Strony www
Wiadomość przychodząca	Nadawca
	Treść wiadomości
Wiadomość wychodząca	Odbiorca
	Treść wiadomości
Spotkanie	Opis spotkania
	Termin
	Sala
	Osoby zaproszone
	Osoby, które potwierdziły
	Osoby, które odmówiły
Plik	Zawartość pliku
	Wersja
	Ścieżka źródłowa

Tab. 2.2: Dane i metadane charakterystyczne dla danego typu informacji. Opracowanie własne.

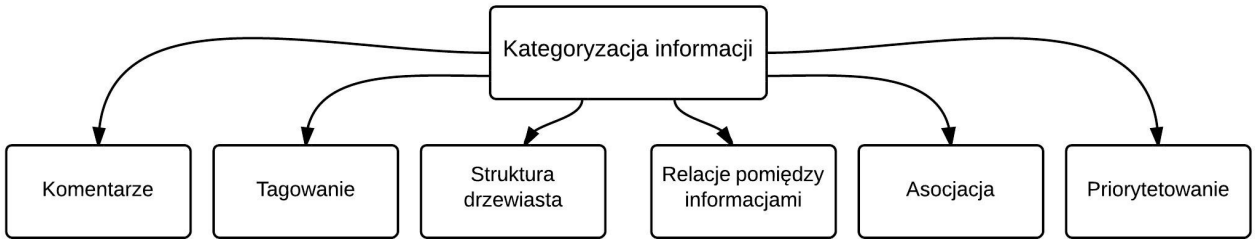
3. Kategoryzacja informacji

System do zarządzania informacją osobistą powinien charakteryzować się jak największą możliwością kategoryzacji informacji, w celu jak najbardziej efektywnego wyszukiwania informacji w przyszłości. Dzięki kategoryzacji informacji, ułatwione jest skuteczniejsze wyszukiwanie informacji relewantnych.

Z powodu szczególnej specyfiki systemów do zarządzania informacją osobistą, podjęto próbę utworzenia autorskiej systematyki, dotyczącej sposobów kategoryzacji informacji w tego typu systemach.

Poniżej zaproponowano następujące sposoby kategoryzacji informacji, jakie powinny być dostępne w systemach do zarządzania informacją osobistą:

- Komentarze
- Tagowanie
- Struktura drzewiasta
- Relacje pomiędzy informacjami
- Asocjacja
- Priorytetowanie



3.1. Komentarz

Komentarz jest jednym z najprostszych sposobów kategoryzacji informacji. Informacja jest opisywana szczegółowo notatką. Przykładowym komentarzem jest opinia użytkownika spisana w formie tekstowej (jako notatka) o wybranym pliku lub grupie plików. Przykładowym zapytaniem użytkownika w systemie do zarządzania informacją osobistą może być znalezienie wszystkich informacji posiadających dane słowo w swoim komentarzu.

3.2. Tagowanie

Rys. 3.1: Chmura tagów opisująca zagadnienia poruszane w serwisie sprawnymarketing.pl

Tagowanie [11] jako sposobem kategoryzacji informacji w systemach do zarządzania informacją osobistą, które można znaleźć również w wielu serwisach komercyjnych (np. sklepach internetowych), społecznościowych jak i również w internetowych katalogach bibliotecznych. Przykładem serwisu internetowego, w którym ten sposób kategoryzacji w pełni się rozwinął i został zaaprobowany przez użytkowników, jest internetowy serwis społecznościowy dla programistów stackoverflow.com. Serwis ten posiada już ponad 35 tysięcy tagów, gdzie kilka najbardziej popularnych z nich, osiągnęło już liczbę ponad 500 tysięcy powiązanych z nimi wątków dyskusji.

Tag oznacza pewne słowo kluczowe, które charakteryzuje jedną lub więcej informacji. Również jedna informacja może posiadać wiele różnych tagów, jednak żadna informacja nie posiada nigdy zdublowanego tego samego tagu. Tagi określa się często "tanimi metadanymi", ponieważ użytkownicy najczęściej sami tworzą sobie w prosty sposób swoje własne tagi, które będą opisywać ich informacje. Korzystanie z tagów jest nie tylko szybkie i elastyczne, ale również umożliwia charakterystykę informacji z wielu perspektyw, włącznie z pełną subiektywizacją opisu. Najczęstszej spotykane rodzaje tagów to:

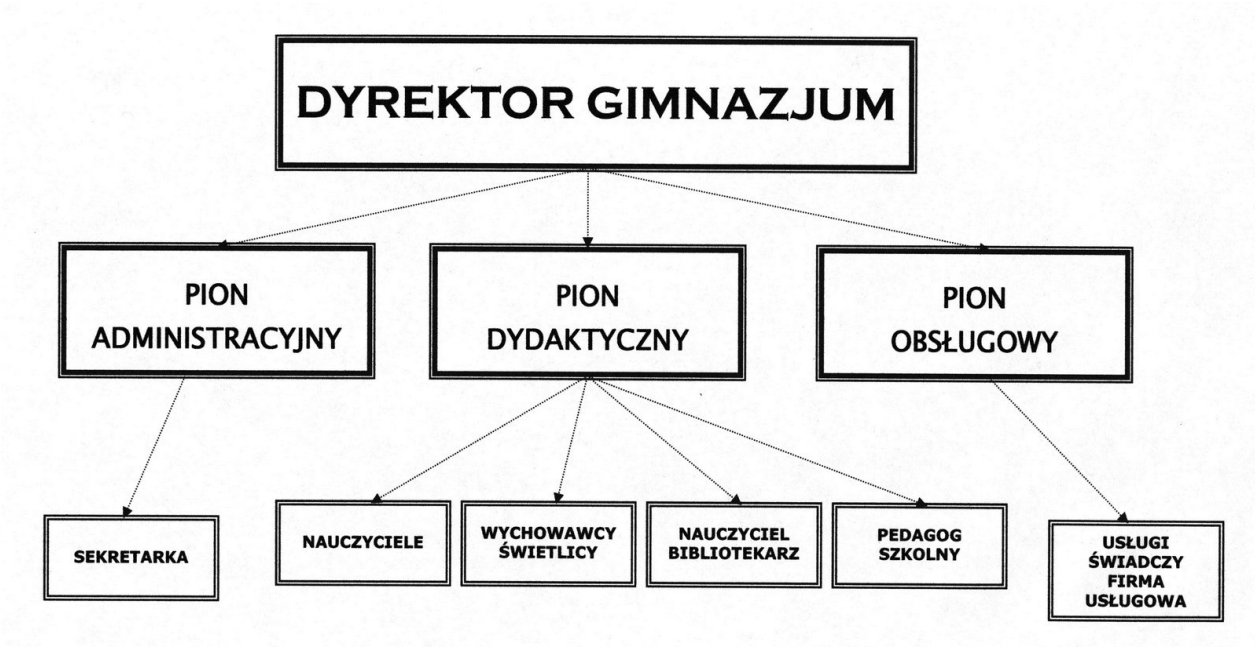
- opisowe - obiektywnie charakteryzujące treść informacji
- formalne - wskazujące na sposób utrwalenia informacji
- własność/źródło - wskazujące na autora, miejsce publikacji
- opinie - wyrażające subiektywne odczucia i nastawienie emocjonalne użytkownika o informacji np. drogi, tani, nudny, interesujący
- auto-odniesienia - np. moje koncerty, książki przeczytane, moje albumy
- organizujące - np. do przeczytania, do zrobienia, do obejrzenia

Rozpatrując tagi, warto zwrócić uwagę na ich liczebność w danym ich zbiorze.

- Zbiór stale się powiększa, ponieważ pojawiają się stale nowe tagi – np. Literatura, Fantastyka, Polityka, Informatyka, Historia, Maraton, Koszykówka
- Zbiór jest generalnie określony, aczkolwiek może być kilka niewielkich modyfikacji oraz uzupełnień w przyszłości np. Zadanie skończone, Zadanie rozpoczęte, Zadanie nierozpoczęte
- Zbiór jest w pełni skończony i niemodyfikowalny np. Kobieta, Mężczyzna

Przykładowym zapytaniem użytkownika w systemie do zarządzania informacją osobistą może być znalezienie wszystkich informacji posiadających ten sam tag (lub grupę tagów) co zadana informacja.

3.3. Struktura drzewiasta



Rys. 3.2: Przykładowa struktura drzewiasta Gimnazjum. Opracowanie własne.

Informacje mogą zostać powiązane ze sobą w strukturę drzewiastą. Drzewa w naturalny sposób reprezentują hierarchię informacji, tak więc w tym celu powinny być głównie stosowane w systemach do zarządzania informacją osobistą. Drzewa składają się z wierzchołków (węzłów) oraz łączących je krawędzi. Jeden z wierzchołków jest wyróżniony i nazywany jest korzeniem drzewa. Wszystkie wierzchołki połączone z danym wierzchołkiem, a leżące na następnym poziomie są nazywane dziećmi tego węzła. Wierzchołek może mieć dowolną liczbę dzieci, jeśli nie ma ich wcale, nazywany jest wtedy liściem. Wierzchołek jest rodzicem dla każdego swojego dziecka. Każdy węzeł ma dokładnie jednego rodzica, wyjątkiem jest korzeń drzewa, który nie ma rodzica.

Podstawowe operacje na drzewach to:

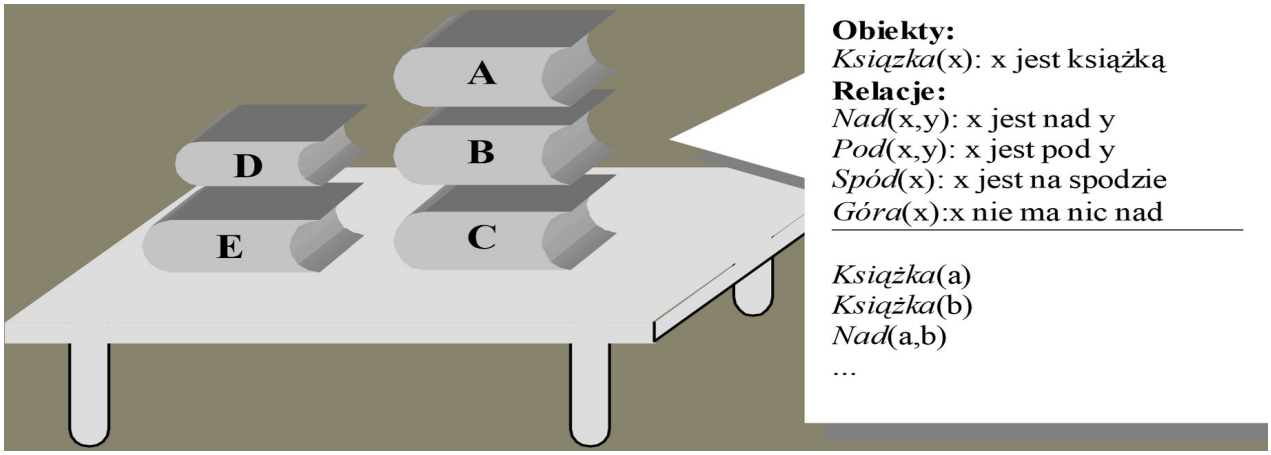
- wyliczenie wszystkich elementów drzewa
- wyszukanie konkretnego elementu
- dodanie nowego elementu w określonym miejscu drzewa
- usunięcie elementu

Struktura drzewiasta w zarządzaniu informacją jest często używana do bardziej szczegółowego opisu struktury danego przedsięwzięcia, organizacji, projektu czy zagadnienia. Struktura podziału pracy (ang. Work Breakdown Structure) jest podstawową technika w zarządzaniu przedsięwzięciami pomagająca określić i zorganizować zasięg przedsięwzięcia przy pomocy hierarchicznej struktury drzewa. Pierwsze dwa poziomy drzewa (korzeń i poziom 2.) określają zbiór oczekiwanych celów przedsięwzięcia, które przedstawiają cały zakres przedsięwzięcia. Na każdym kolejnym poziomie węzły w sumie powinny reprezentować pełny zakres węzła nadrzędnego.

Przykładowym zapytaniem użytkownika w systemie do zarządzania informacją osobistą może być znalezienie wszystkich informacji, które są dziećmi danego węzła (zadanej informacji).

3.4. Relacje pomiędzy informacjami

Aby uporządkować wiedzę o relacjach między informacjami, potrzebna jest metoda indeksowania i klasyfikacji. Informacje można zobrazować za pomocą ontologii, czyli schematu przedstawiającego składniki i ich wzajemne powiązania. Ontologia ułatwia porządkowanie i pojmowanie informacji oraz zarządzanie nimi. Ontologia jest formalną specyfikacją wspólnej warstwy pojęciowej, a więc dostarcza formalnego słownictwa do opisania danej dziedziny. W ontologii między pojęciami mogą występować powiązania o nietaksonomicznym charakterze, na przykład relacje typu „zawiera". Systemy do zarządzania informacją osobistą powinny mieć możliwość definiowania ontologii, czyli formalnej reprezentacji wiedzy, na którą składa się zbiór informacji oraz relacji między nimi [12].



Rys. 3.3: Przykładowa ontologia. [12]

W. Gliński definiuje przykładową ontologię dla rys. 3.3 następująco: [12]

Obiekty
Istnieje 5 obiektów tego samego typu, które są oznaczone jako a,b,c,d,e.
Książka(x) oznacza: x jest książką

Podstawiając pod "x" nazwę książki, uzyskujemy
Książka(a), Książka(b), Książka(c), Książka(d), Książka(e).

Relacje
Binarne:
- Nad(x,y): x jest nad y
- Pod(x,y): x jest pod y

Unarne:
- Spód(x): x jest na spodzie
- Góra(x): x jest na górze

Przykładowym zapytaniem użytkownika w systemie do zarządzania informacją osobistą może być znalezienie wszystkich informacji, które występują w danej relacji do zadanej informacji. Przykładowo w języku UML, agregacja jest często określana jako relacja typu "zawiera" np. "samochód zawiera silnik".

3.5. Asocjacja

Asocjacja jest najprostszym i najczęściej występującym sposobem powiązywania informacji. Nie definiuje one żadnego kontekstu dla dwóch różnych informacji, a jedynie informuje, że dwie różne informacje są ze sobą w jakiś sposób powiązane. Asocjację w ujęciu ontologicznym można przedstawić jako relację typu „jest powiązana" np. „informacja pierwsza jest powiązana z informacją drugą".

3.6. Priorytetowanie

Priorytetowanie informacji oznacza ustalenie pewnej określonej kolejności, relacji porządku dla podanych informacji. Może być ona zdefiniowania w formie ontologii, jak w podrozdziale 3.4, dla przykładu:
Przed(x,y) – x jest przed y
Jednak tak zdefiniowana relacja binarna może doprowadzić do pewnych sprzeczności.

Przykład:
Istnieją 3 informacje, które oznaczone są jako a,b,c.

Użytkownik nadał tym informacjom następujące relacje:
Przed(a,b), Przed(b,c), Przed(c,a)

Tak zdefiniowana relacja nie pozwala uwzględnić przechodniości tej relacji, ponieważ w przeciwnym przypadku, informacja A była by przechodnio przed informacją C, co jest sprzecznością, ponieważ informacja C jest przed informacją A.

Nie ulega wątpliwości, że aby prawidłowo dokonać priorytetowania informacji, należy powiązać każdą informację tylko z jednym z elementów będącym w zbiorze mającym już prawidłowo zdefiniowaną relację porządku. Przykładem takiego zbioru jest lista określeń używana przez programistów do oszacowania surowości (ang. severity) błędu programistycznego.

Poniżej została zaprezentowana przykładowa powyższa lista uszeregowana według odpowiedniej kolejności, od błędu najpoważniejszego do najmniej ważnego:

- Błąd blokujący (ang. blocker)
- Błąd krytyczny (ang. critical)
- Błąd poważny (ang. major)
- Błąd pomniejszy (ang. minor)
- Błąd trywialny (ang trivial)

Warto podkreślić, że również doskonałym zbiorem mającym już prawidłowo zdefiniowaną relację porządku, jest zbiór liczb całkowitych. Poprzez powiązanie każdej informacji z dokładnie jedną liczbą należącą do zbioru liczb całkowitych, uzyskujemy prawidłowo zdefiniowaną relację porządku dla tak zdefiniowanego zbioru informacji.

Przykładowym zapytaniem użytkownika w systemie do zarządzania informacją osobistą może być znalezienie wszystkich informacji z danego zbioru, które posiadają większy priorytet niż zadana informacja.

4. Problemy informacyjne w zarządzaniu informacją osobistą

Zarządzanie informacją osobistą stawia przed wieloma wyzwaniami. Niewłaściwe zarządzanie informacjami przez użytkownika doprowadzi do zalegania niepotrzebnych informacji jak i przeciążenia informacyjnego, które nieefektywnie wpłynie na późniejszą pracę, wydłużając czas potrzebny na wyszukiwanie informacji lub nawet uniemożliwiając wykonanie pewnych zadań. Ponadto, każdy z użytkowników rozpoczynający prace z systemem, stoi przed problemem braku jakichkolwiek informacji wprowadzonych do systemu. Niniejszy rozdział jest poświęcony próbie zdefiniowania podstawowych problemów, z którymi zetknie się każdy użytkownik, korzystający z systemów do zarządzania informacją osobistą. Każdy z opisanych niżej problemów wpłynie negatywnie na skuteczne wyszukiwanie i rekomendowanie informacji relewantnej dla użytkownika, dlatego bardzo ważna jest znajomość tych problemów i odpowiednich technik przeciwdziałających im.

4.1. Informacje odseparowane

Problem nowej informacji

Jednym z najczęstszych problemów jest problem nowej informacji. Nowe dane nie są dobre pod względem wymiarów jakości danych, dostępności, aktualności oraz relewantności, ponieważ z powodu żadnego powiązania z innymi informacjami, użytkownik ma bardzo małe szanse na dotarcie do nich. W takim wypadku użytkownik ma możliwość skorzystania z 2 możliwych rozwiązań**:**

- **Ręczne powiązanie informacji**

  Użytkownik może ręcznie powiązać inne informacje z daną informacją. To rozwiązanie jest stosowane najczęściej. Użytkownik niestety nie zawsze jest w stanie odtworzyć samemu wszystkie powiązania pomiędzy informacjami. Jednak większość najważniejszych informacji dla użytkownika, zostanie przez niego osobiście dodana do systemu i powiązana z zadaną informacją.

- **Stereotypowanie**

  Stereotypowanie polega na inteligentnym oraz automatycznym przetworzeniu informacji, oraz wykryciu pewnych zależności z innymi informacjami na podstawie podobieństwa zadanej informacji do innych informacji. Niektóre takie procesy takie mogą być bardzo czasochłonne, zatem dobrym rozwiązaniem jest uruchomienie ich tylko na wyraźne żądanie użytkownika. Tak zdefiniowany problem decyzyjny, opierający się na wyszukaniu pewnych zależności, jeżeli został rozwiązany pozytywnie, powinien zarekomendować rozwiązanie użytkownikowi, które ten może opcjonalnie przyjąć (z dodatkową możliwością wprowadzenia poprawek do zaproponowanego rozwiązania). Bardzo istotne jest to, aby unikać rekomendacji fałszywie pozytywnych, które zostaną opisane szerzej w rozdziale 5.

Problem słabo powiązanej informacji

Problem słabo powiązanej informacji jest bardzo podobny do problemu nowej informacji. Również w tym przypadku, możliwymi do zastosowania rozwiązaniami są stereotypowanie oraz ręczne powiązanie informacji. Jednak korzystając w tym przypadku z stereotypowania, algorytm wyszukujący pewne zależności, posiada teraz większą wiedzę na podstawie już powiązanych informacji.

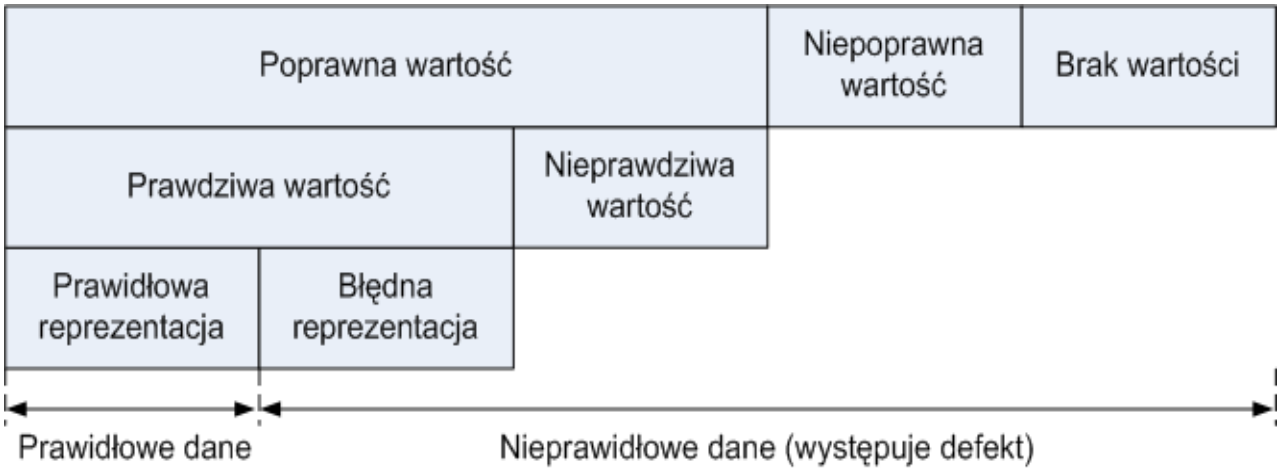
Problem całkowicie odseparowanej informacji

Problem ten jest również podobny do problemu nowej informacji. Informacja w systemie jest całkowicie odseparowana od innych informacji, tak więc posiada właściwości takie same jak informacja, która została najpóźniej wprowadzona do systemu. Również w tym przypadku możliwymi rozwiązaniami są stereotypowanie oraz ręczne powiązanie informacji.

4.2. Informacje nieprawidłowe i niezweryfikowane

Informacje są danymi zinterpretowanymi przez odbiorcę. Analizując te dane w aspekcie ich jakości, można podzielić je według rodzaju problemów w nich występujących: [13]

- **dane prawidłowe** (np. mieszkaniec Wrocławia wskazał „Wrocław” jako miejsce zamieszkania)
- **dane poprawne, ale nieprawdziwe** (np. mieszkaniec Wrocławia wskazał „Poznań” jako miejsce zamieszkania)
- **dane niepoprawne** (np. mieszkaniec Wrocławia wskazał „Nibylandia II” jako miejsce zamieszkania).



Rys. 4.1: Klasyfikacja danych w kontekście występujących w nich problemów. [13]

Problem niepełnych informacji

Problem ten jest jednym z najczęściej występujących problemów w systemach do zarządzania informacją osobistą. Informacje, które nie są w pełni uzupełnione, prowadzą do problemu tzw. anemii informacyjnej.

Problem nieprecyzyjnych informacji

Problem ten pojawia się podczas przypisania prawidłowej, ale niewystarczająco szczegółowej informacji. Przykładem jest podanie jedynie pierwszej litery imienia mającego więcej niż jedno rozwinięcie („P.” może oznaczać Piotra, Pawła, Patryka, itd.). Problem ten może również wpłynąć na powstanie problemu dwuznaczności informacji.

Problem starzejących się informacji

Zmiany zachodzące w rzeczywistym świecie, które nie są odnotowywane (aktualizowane), są przyczyną starzenia się informacji. Jest to czynnik zewnętrzny, ale ma on bezpośrednio wpływ na jakość i wartość zgromadzonych danych w systemie do zarządzania informacją osobistą.

Problem sprzecznych informacji

Użytkownik może podać kilka ze sobą sprzecznych informacji. Użytkownik może również podać kilka informacji, które będąc rozpatrywane każde z osobna, są prawdziwe, jednak rozpatrywane wszystkie jednocześnie w całym zbiorze informacji są ze sobą sprzeczne.

Problem braku automatycznej weryfikacji informacji

Systemy do zarządzania informacją osobistą, są przykładem systemów informacyjno-wyszukiwawczych, i w przeciwieństwie do systemów ewidencyjnych, nie posiadają generalnie możliwości walidacji wprowadzonych danych, tak jak jest to rygorystycznie przestrzegane w systemach ewidencyjnych. Wprawdzie istnieją najprostsze, predefiniowane przez twórców systemów, walidacje podstawowych typów informacji (np. wiek osoby w informacji typu „kontakt" nie może być mniejszy od zera), jednak na tym kończą się możliwości, z których może skorzystać użytkownik. Użytkownik, nie będący programistą, nie może zdefiniować bardziej wymagających ograniczeń na swoje informacje – przykładowo, aby system dopuszczał tylko utworzenie nowego lub zmodyfikowanie starego kontaktu, który będzie odpowiadał osobie posiadającej pełnoletniość oraz miejsce zamieszkania zdefiniowane jako „Wrocław" lub „Opole". Z tego powodu, ciężar odpowiedzialności za weryfikację informacji spada całkowicie na użytkownika systemu.

Problem braku możliwości definiowania reguł biznesowych

Problem ten jest powiązany z wcześniej wspomnianym problemem, wynikającym z faktu, że systemy do zarządzania informacją osobistą, nie są typowymi systemami ewidencyjnymi, które narzucają integralność danych i wymuszają spełnienie zaawansowanych reguł biznesowych. Przeprowadzony przegląd rynku wskazuje, że systemy do zarządzania informacją osobistą nie oferują możliwości tworzenia własnych typów informacji wraz z zależnościami oraz ograniczeniami, które te informacje muszą spełniać. Przykładowo, użytkownik nie może zdefiniować ograniczenia, którego skutkiem było by uniemożliwienie wprowadzenia do systemu zdjęcia, utworzonego w XX wieku, posiadającego jednocześnie tag „Dziecko", a odnoszącego się do dziecka, które przyszło na świat dopiero w XXI wieku.

Problem braku spójności informacji

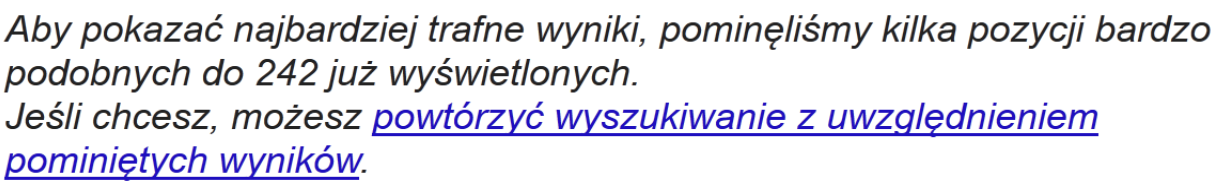
Problem ten jest związany z synonimami, czyli możliwością różnorodnego zdefiniowania tej samej informacji. Przyczyna leży zarówno w braku dyscypliny użytkownika, jak i braku weryfikacji wprowadzanych informacji. Użytkownik może zarówno opisać „poniedziałek" jako „1 dzień tygodnia" lub „PN". Podobnie termin „auto" można również zastąpić terminem „pojazd". Problem ten może również wpłynąć na powstanie problemu zdublowanych informacji, opisanego w następnym punkcie. Poniżej przedstawiono przykład takiego problemu, powstałego z przyczyny braku zachowania spójności przez użytkownika przy wprowadzaniu danych.

GÓRZNO
GRABÓW
GRABÓW N PROSNĄ
GRABÓW N/ PROSNĄ
GRABÓW N/PR
GRABÓW N/PR.
GRABÓW N/PROSNA
GRABÓW N/PROSNĄ
GRABÓW N\PROSNĄ
GRABÓW NAD PROSNĄ
GRĘBANIN

Tab. 4.1: Duplikaty w słowniku miejscowości

Problem zdublowanych informacji

Problem zdublowanych informacji występuje również w wyszukiwarkach internetowych. Wyszukiwarki internetowe chcą zaproponować użytkownikowi zróżnicowaną grupę wyników w przeciwieństwie do wielu setek stron z dokładnie tą samą treścią. Przykładowo, wyszukiwarka Google pomija najbardziej podobne wyniki i oferuje użytkownikowi możliwość wyświetlenia zdublowanych wpisów na żądanie:



Rys. 4.2: Informacja o pominięciu zdublowanych wyników w wyszukiwarce Google

Problem dwuznaczności informacji

Dwuznaczność, lub wieloznaczność informacji, występuje w sytuacji możliwości różnej interpretacji tej samej informacji, gdy nie da się ustalić, która z nich jest właściwa. W skrajnych przypadkach problem ten może prowadzić do dystorsji informacyjnej. Najczęściej przyczyna tego problemu leży w podaniu nieprecyzyjnych informacji.

5. Rekomendacje

5.1. Klasyfikacja rekomendacji

Zapytania o interesujące użytkownika informacje w systemach do zarządzania informacją osobistą wiążą się mocno z tematem rekomendacji. Rekomendacja w systemach do zarządzania informacją osobistą jest poleceniem pewnej informacji. System proponuje użytkownikowi pewne informacje na podstawie kryteriów, jakie użytkownik zdefiniował. Ilość wyszukanych informacji przez użytkownika może być niejednokrotnie zbyt wielka lub zbyt mała dla użytkownika. Ponadto treść prezentowanych informacji może odbiegać od tego, czego użytkownik naprawdę szukał. Dlatego bardzo istotnym aspektem jest to, aby system prezentował te informacje użytkownikowi, które go rzeczywiście interesują. Dla dobra użytkownika, najważniejsze jest unikanie fałszywie pozytywnych rekomendacji. Poniżej zdefiniowano podstawowy podział rekomendacji [11]:

Rekomendacja jest prawdziwie pozytywna (TP, ang. True Positive), jeżeli wygenerowana rekomendacja odpowiada użytkownikowi. Nazywana jest ona również „poprawnym trafieniem".

Rekomendacja jest prawdziwie negatywna (TN, ang. True Negative), jeżeli rekomendacja nie została polecona użytkownikowi i faktycznie nie odpowiadała ona użytkownikowi. Nazywana jest ona również „poprawnym odrzuceniem".

Rekomendacja jest fałszywie pozytywna (FP, ang. False Positive), jeżeli rekomendacja została wygenerowana, ale nie odpowiadała użytkownikowi. Nazywana jest ona również błędem pierwszego rodzaju lub „fałszywym alarmem".

Rekomendacja jest fałszywie negatywna (FN, ang. False Negative), jeżeli rekomendacja nie została polecona użytkownikowi, mimo że odpowiadała by użytkownikowi. Nazywana jest ona również błędem drugiego rodzaju lub „nie trafieniem".

5.2. Rekomendacje w kontekście systemów do zarządzania informacją osobistą

Rekomendacje są szeroko znane z popularnych portalów społecznościowych jak Facebook, MySpace czy LinkedIn. Dla użytkownika oczywistą korzyścią ze stosowania technik rekomendacji w systemie do zarządzania informacją osobistą jest szybki dostęp do poszukiwanych informacji, zgodnej z preferencjami i oczekiwaniami. Jedną z technik rekomendacyjnych, stosowanych w serwisach społecznościowych jest filtrowanie kolaboratywne.

Serwisy te używają kolaboratywnego filtrowania do np. rekomendacji nowych znajomych, grup (poprzez analizę sieci powiązań pomiędzy użytkownikiem a jego znajomymi). Przykładowo, jeżeli pierwszy użytkownik posiada wielu tych samych, wspólnych znajomych co drugi użytkownik (ich profil jest podobny), bardzo możliwe jest, że również ci dwaj użytkownicy się znają lub lubią podobne rzeczy, dzięki czemu możliwe jest wygenerowanie odpowiednich rekomendacji [14].

Oczywistą zaletą techniki filtrowania kolaboratywnego jest fakt, że nie bazuje ona na analizie konkretnych obiektów, dlatego zdolna jest do rekomendowania różnego rodzaju obiektów, bez faktycznej, szczegółowej wiedzy o danych obiektach.

Sposób rekomendacji w systemie do zarządzania informacją osobistą zasadniczo odbiega od rekomendacji stosowanych w serwisach społecznościowych. Jedną z głównych podstaw filtrowania kolaboratywnego, występującego często w serwisach społecznościowych, jest współpraca użytkowników poprzez udzielanie opinii na temat danych obiektów, aby pomóc sobie nawzajem w wyłonieniu istotnych dla siebie informacji. Zazwyczaj w tym celu stosuje się tzw. profilowanie jawne (ang. explicit profile), przykładowo poszczególne obiekty są prezentowane użytkownikom do oceny, poddawane są głosowaniu, lub np. istnieje możliwość dodania elementu do grupy tzw. ulubionych. Dzięki temu systemy rekomendują te obiekty (informacje) użytkownikowi, które zostały pozytywnie ocenione przez użytkowników o profilach podobnych do profilu danego użytkownika. Jak opisano wcześniej, istnieje możliwość rekomendacji obiektów bez zebrania wiedzy na temat cech czy atrybutów danych obiektów. Niestety technika ta, oparta na korelacji pomiędzy użytkownikami, będzie nieprzydatna w systemach do zarządzania informacją osobistą. Jak wynika z definicji, system taki przechowuje osobiste informacje danego użytkownika, zorientowane wyłącznie tylko do jego własnych, prywatnych potrzeb. Również poddawanie użytkownikowi poszczególnych obiektów (informacji) do oceny lub głosowania pod względem przydatności dla użytkownika mija się z celem. Informacje zawarte w systemie do zarządzania informacją osobistą już są generalnie odpowiednio wartościowe dla użytkownika, ponieważ je sam odpowiednio wcześniej wyselekcjonował i wprowadził do systemu.

Oczywiście zarówno dla serwisów społecznościowych korzystających z metod filtrowania kolaboratywnego jak i systemów do zarządzania informacją osobistą, wspólnym problemem jest problem tzw. zimnego startu. Bez odpowiedniej ilości danych nie jest możliwe dokładne i skuteczne rekomendowanie informacji. Również wspólnym problemem jest ograniczona analiza zawartości. System nie jest w stanie w pełni przeanalizować zawartości obiektu, tak więc musi sprowadzić obiekt do odpowiedniej postaci, która będzie umożliwiała jego dalszą analizę. Identyczny zbiór cech dwóch różnych obiektów może spowodować niemożliwość ich rozróżnienia. Wspólnym problemem jest również analiza zawartości niektórych obiektów takich jak obrazy, dźwięki czy strumienie video. W systemach do zarządzania informacją osobistą nie występuje jednak problem skalowalności, gdyż nie ma potrzeby posiadania tak ogromnej mocy obliczeniowej do przetworzenia milionów użytkowników, produktów i innych obiektów w celu wyliczenia odpowiedniej rekomendacji dla użytkownika. Ponadto, system do zarządzania informacją osobistą jest wolny od problemu oszukiwania systemu rekomendacyjnego, występującego często w kolaboratywnych systemach rekomendacyjnych. Oszustwa te zazwyczaj służą zwiększeniu popularności własnych przedmiotów. Do powszechnie znanych takich praktyk należy zatrudnianie użytkowników wysoko oceniających wskazane przedmioty oraz wykorzystywanie botów internetowych generujących sztuczne zainteresowanie wskazanymi obiektami.

Dużo bardziej zbliżonym sposobem rekomendacji w systemach do zarządzania informacją osobistą w stosunku do systemów wspierających pracę w środowisku wieloużytkownikowym, jest rekomendacja w oparciu o treść. W systemach rekomendujących w oparciu o treść podstawą do wyznaczania podobieństwa pomiędzy obiektami w systemie są stowarzyszone z nimi cechy. Każdy obiekt jest reprezentowany przez zbiór deskryptorów lub termów, najczęściej jest to częstotliwość słów występujących w dokumencie. Algorytm musi mieć możliwość spójnego porównania różnych obiektów na podstawie tych samych termów. W systemach do zarządzania informacją osobistą, najczęściej źródłem do rekomendacji obiektów (informacji) są dane tekstowe. Reprezentacją dla danych tekstowych jest najczęściej model przestrzeni wektorowej, które zostanie szczegółowo opisany w następnym rozdziale.

6. Wyszukiwanie informacji

Sama kategoryzacja informacji bezpośrednio nie definiuje, które informacje są dla siebie relewantne. Ponieważ relewancja informacji jest wartością subiektywną, zależną od użytkownika, użytkownik podczas wyszukiwania informacji powinien móc określić jakie informacje go rzeczywiście interesują. Celem użytkownika pracującego ze swoimi prywatnymi informacjami, jest jak najłatwiejsze zarządzanie nimi. Jego głównymi zadaniami są wyszukiwanie informacji już istniejących w systemie, aktualizacja starych informacji oraz dodawanie nowych. Specyfika informacji użytkownika, przechowywanych w systemie do zarządzania informacją osobistą, mocno się różni od informacji dostępnych w sieci WWW. W dobie internetu zalewającego nas ogromną ilością informacji, tylko niewielka część informacji znaleziona przez typowe wyszukiwarki internetowe jest dla użytkownika wartościowa. Z tego powodu informacje przechowywane przez system do zarządzania informacją osobistą są zazwyczaj dużo cenniejsze dla użytkownika, ponieważ zostały one osobiście przez użytkownika wyselekcjonowane i wprowadzone do systemu. Jego informacje często są pogrupowane w sposób tematyczny, najbardziej przystępny dla użytkownika. Bardzo często część danych w systemie do zarządzania informacją osobistą jest strukturyzowana, co upraszcza dodatkowo wyszukiwanie informacji w systemie, poprzez stosowanie różnych kryteriów wyszukiwania. Liczba informacji jest również nieporównywalnie mniejsza od informacji, które można wyszukać w internecie. Dodatkowo informacje występujące w sieci WWW nie mają jednolitej struktury (znanej użytkownikowi). Oczywiście wspólnym problemem dla wszystkich informacji jest fakt, że wszystkie informacje stopniowo w czasie podlegają dezaktualizacji.

Zakładając, że użytkownik wprowadził wcześniej odpowiednie informacje do systemu, głównym problemem dla użytkownika jest tylko wyszukanie odpowiednich informacji. Większa część informacji w systemie do zarządzania informacją osobistą nie jest strukturyzowana (najczęściej dokumenty tekstowe) lub częściowo strukturyzowana (przykładowo dokumenty tekstowe z paroma zdefiniowanymi polami – np. tytuł i rok powstania). Z tego powodu niezbędne jest wypracowanie technik, które pozwolą na odpowiedzenie na potrzebę informacyjną użytkownika właśnie dla tego typu danych. Przykładowym wyzwaniem w tej dziedzinie problemów jest znalezienie wszystkich wiadomości mailowych wysłanych do wskazanego adresata, które mają związek tematyczny z daną wiadomością zdefiniowaną przez użytkownika. O ile wyszukanie wszystkich wiadomości mailowych wysłanych tylko do danego adresata jest zadaniem trywialnym, to przefiltrowanie ich pod względem relewantności dla wskazanego zapytania jest trudnym wyzwaniem.

Aby skutecznie operować na danych niestrukturyzowanych (tekstowych) i odpowiadać na tego rodzaju powyższe potrzeby, należy sięgnąć w tym celu do osiągnięć z dziedziny wyszukiwania informacji (ang. information retrieval), operującej w głównej mierze na niestrukturyzowanych danych tekstowych. Osiągnięcia z dziedziny wyszukiwania informacji, mającej już ponad 50 lat aktywnej działalności, doprowadziły do powstania wiele modelów służących odpowiedniemu wyszukiwaniu informacji dla zbioru dokumentów tekstowych. Poniżej zaprezentowano podstawowe modele do reprezentacji tekstu i zapytań [15]:

- Modele oparte o zbiór słów kluczowych
- Modele oparte o reprezentację wektorową
- Modele probabilistyczne

Do najczęstszych zadań z dziedziny wyszukiwania informacji należy wyszukiwanie dokumentów relewantnych w oparciu o zapytanie użytkownika (zbiór słów kluczowych) lub w oparciu o przykładowy dokument.

6.1. Model boolowski

Jednym z modeli opartych o zbiór słów kluczowych jest model boolowski, którego zapytanie ma postać kwerendy logicznej [15]. Kwerenda logiczna zbudowana jest z termów połączonych spójnikami logicznymi AND, OR i NOT. Definicja termów zostanie szczegółowo omówiona w podrozdziale 6.7. Term jest uznawany za prawdziwy, jeżeli wyraz występuje w dokumencie. Pod definicją dokumentów, przyjmujemy dowolne, podstawowe elementy systemu przechowujące informacje, na których operuje cały system do zarządzania informacją osobistą. Proces wyszukiwania informacji w modelu boolowskim polega na wybieraniu z kolekcji tych dokumentów, dla których kwerenda jest prawdziwa. Z tego powodu model boolowski jest często nazywany modelem dokładnego dopasowania (ang. exact match) [15].

Rozpatrzmy binarną macierz incydencji term-dokument (tab. 6.1), w której zostanie zaznaczone odpowiednio występowanie danego termu (lub jego brak) w podanej sztuce z dzieł Szekspira.

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

Tab. 6.1: Macierz incydencji term-dokument. Element macierzy o wierszu *i* i kolumnie *j* jest równy 1, gdy dany dokument z kolumny j zawiera term *i*. [15]

Rozpatrując powyższą macierz incydencji według kolumn, uzyskujemy wiedzę z jakich termów składa się każdy dokument. Rozpatrując powyższą macierz incydencji według wierszy, uzyskujemy wiedzę, w jakich dokumentach występuje dany term.

Kwerenda logiczna, pokazująca w jakich dokumentach występują jednocześnie termy "Brutus" i "Caesar" i nie występuje jednocześnie "Calpurnia", wygląda następująco:

Brutus AND *Caesar* AND NOT *Calpurnia*

Wykonując operację iloczynu logicznego (po uprzednim utworzeniu dopełnienia logicznego dla dokumentu *Calpurnia*), uzyskujemy:

110100 AND 110111 AND 101111 = 100100

Wynikiem powyższej kwerendy logicznej, odpowiadającej na pytanie w jakich dokumentach występują jednocześnie termy "Brutus" i "Caesar" i nie występuje jednocześnie "Calpurnia", jest zbiór:
"Antony and Cleopatra" oraz "Hamlet"

6.2. Problemy związane z modelem boolowskim

Model boolowski jest dobry dla doświadczonych użytkowników, dobrze rozumiejących swoje potrzeby, dziedzinę wyszukiwania i bazę dokumentów. Niestety jest nieodpowiedni dla większości użytkowników, którzy nie potrafią sformułować swoich potrzeb w postaci wyrażeń logicznych. Strategia wyszukiwawcza modelu boolowskiego bazuje na binarnym kryterium decyzyjnym – dokument albo jest relewantny, albo nie, bez żadnej pośredniej gradacji. Może to pogarszać jakość działania systemu, zwłaszcza biorąc pod uwagę nieprecyzyjny charakter języka naturalnego. Eliminowanie z listy wyników odpowiedzi tych dokumentów, które w pełni precyzyjnie nie odpowiadały kwerendzie (choć spełniły większość z warunków), pogarsza rezultaty wyszukiwania [16].

Problemem modelu boolowskiego jest również ilość zwracanych wyników.
Użytkownik chce przejrzeć co najwyżej kilka rezultatów w przeciwieństwie do kilku tysięcy. Jest bardzo trudnym, nawet dla ekspertów, aby przy pomocy odpowiedniej kwerendy wyszukać odpowiednią ilość wyników. Bardzo często na podstawie danego zapytania, znalezionych dokumentów jest za dużo lub kryterium jest zbyt restrykcyjne i żaden dokument nie jest zwracany użytkownikowi. [15]

Oczywistym jest, że większość z wyszukanych informacji różni się relewancją w stosunku do danego zapytania. Prezentowanie informacji użytkownikowi według określonego porządku zgodnie z wyliczonym poziomem relewancji, od największego do najmniejszego, jest bardziej efektywne i użyteczne. Z powodu znacznej ilości możliwych wyników wyszukiwania, posiadanie rankingu wyszukanych wyników jest niezbędne. Wyszukiwanie przy pomocy modelu boolowskiego zwraca wyniki bez żadnego określonego porządku (rankingu), co jest jedną z największych wad tego modelu.

Pomimo swej prostoty, jasnego formalizmu i precyzyjnych zapytań, nie jest łatwo przełożyć na wyrażenia boolowskie własną potrzebę informacyjną. Większość, zwłaszcza nieprofesjonalnych użytkowników uważa wyrażanie swoich zapytań w postaci wyrażeń boolowskich za trudne. Ograniczają się do najprostszych warunków, które nie pozwalają na właściwy opis potrzeb informacyjnych. [16]

Jak wspomniano powyżej, model boolowski ma swoje zalety i wady, i choć spełni on swoje założenia, to jednak taki sposób wyszukiwania informacji dla użytkownika nie będzie w pełni wystarczający.

Rozpatrzmy dwa przypadki systemów przechowujących informacje, gdzie jedne z najważniejszych metainformacji to są występujące zależności (asocjacje) pomiędzy obiektami (informacjami). Rozpatrzmy portal społecznościowy oraz bardzo rozbudowaną stronę WWW. W przypadku portalu społecznościowego przykładowymi obiektami są poszczególne osoby, a znajomość pomiędzy konkretnymi dwoma osobami jest przykładem asocjacji. W przypadku strony WWW, wszystkie artykuły są przykładowymi obiektami a prowadzące w nich do innych artykułów hiperłącza są przykładowymi asocjacjami. Przykładowym problemem w obu systemach będzie zapytanie: "co łączy dwa wskazane obiekty ze sobą?". Samo wyszukanie dwóch różnych użytkowników w portalu społecznościowym korzystając z modelu boolowskiego nie rozwiązuje jeszcze powyższego problemu. Identycznie w przypadku strony WWW, wyszukanie przy pomocy odpowiednich termów najpierw pierwszego artykułu, a następnie drugiego artykułu, nie odpowiada również na powyższe pytanie, jakie inne artykuły mają związek z wyszukanymi dwoma artykułami. Problem ten może zostać lepiej rozwiązany z użyciem modelu przestrzeni wektorowej, który w przeciwieństwie do modelu boolowskiego nie jest modelem dokładnego dopasowania, lecz modelem najlepszego dopasowania (ang. best match). [15]

6.3. Model przestrzeni wektorowej

Model przestrzeni wektorowej ma (w przeciwieństwie do boolowskiego) charakter algebraiczny, a nie logiczny, można zatem łatwo zrezygnować z ograniczających warunków w postaci binarnej, prawdy lub fałszu. Model ten zatem został stworzony w odpowiedzi na problemy wynikające z modelu boolowskiego, będącego przykładem modelu dokładnego dopasowania. W modelu przestrzeni wektorowej, zapytanie i dokumenty reprezentowane są jako wektory

$$\mathbf{q} = [q_1, q_2, ..., q_n]$$
$$\mathbf{d}_j = [w_{1j}, w_{2j}, ..., w_{nj}]$$

w przestrzeni R^n, gdzie n jest liczbą wszystkich termów indeksujących (rozmiar słownika) [16]. Według autorów [15], system SMART był najprawdopodobniej pierwszym systemem, który zdefiniował model reprezentacji dokumentów w postaci wektora wag termów. Term i w opisie dokumentu j, reprezentowany jest przez pewną nieujemną liczbę rzeczywistą w_{ij}, nazywaną „wagą". Im większa wartość wagi termu, tym bardziej jest on istotny dla opisu treści dokumentu. Wartość $w_{ij} = 0$ oznacza, że dany term w opisie dokumentu nie występuje.

W tab. 6.2 waga termu oznacza ilość wystąpień danego termu w danym dokumencie.

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
ANTHONY	157	73	0	0	0	1
BRUTUS	4	157	0	2	0	0
CAESAR	232	227	0	2	1	0
CALPURNIA	0	10	0	0	0	0
CLEOPATRA	57	0	0	0	0	0
MERCY	2	0	3	8	5	8
WORSER	2	0	1	1	1	5

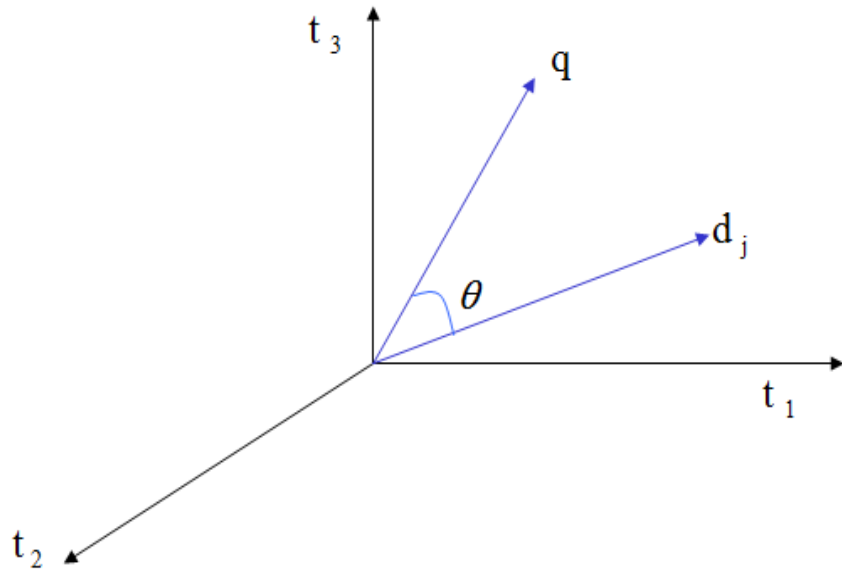
Tab. 6.2: Macierz występowania term-dokument. [15]

W modelu przestrzeni wektorowej, każdy dokument jest reprezentowany przez wektor przestrzeni n-wymiarowej. Oczywiście przestrzeń ta jest bardzo duża ze względu na ilość możliwych rozpatrywanych termów we wszystkich dokumentach w kolekcji. Większość wartości w każdym z wektorów jest równa zero, co oznacza, że macierz jest rzadka (ang. sparse matrix). Zapytanie użytkownika również można przedstawić w postaci wektora przestrzeni n-wymiarowej. Podobieństwo wektora zapytania do wektora danego dokumentu prawdopodobnie oznacza, że dany dokument jest relewantny dla danego zapytania. Zatem istnieje potrzeba zdefiniowania funkcji oceny miary podobieństwa (ang. similarity) danego wektora zapytania (lub wektora innego dokumentu) do wektora danego dokumentu. Sama norma euklidesowa nie jest odpowiednia do zastosowania jako miara podobieństwa. Dla dokumentów nawet o podobnej tematyce i podobnym rozkładzie termów, lecz o różnej długości, norma euklidesowa jest bardzo duża. W tym celu należy dokonać normalizacji wektorów, poprzez podzielenie każdej składowej wektora przez długość całego wektora. [15]

Po dokonaniu normalizacji wektorów, dokumenty o podobnej tematyce powinny charakteryzować się podobną wartością wag występowania identycznych termów, nawet jeżeli te dokumenty są różnej długości. Jak wspomniano wcześniej, dokumenty, które są „blisko" siebie w przestrzeni wektorowej, dotyczą najprawdopodobniej tej samej problematyki i opisują te same zagadnienia. Aby porównać podobieństwo (stopień relewancji), należy zastosować miarę kosinusoidalną. Miara kosinusoidalna reprezentuje kosinus kata pomiędzy dwoma wektorami reprezentującymi dokumenty– znormalizowany iloczyn skalarny wektorów [16]:

$$sim(q,d_j) = \frac{\sum_{i=1}^{n} q_i w_{ij}}{\sqrt{\sum_{i=1}^{n} q_i^2 \sum_{i=1}^{n} w_{ij}^2}} = \frac{q \cdot d_j}{|q||d_j|}$$

Powyższa miara przyjmuje wynik z zakresu [0,1]. Licznik w powyższym równaniu oznacza iloczyn skalarny wektora zapytania *q* oraz wektora dokumentu *j*, mianownik oznacza iloczyn normy euklidesowej wektora zapytania q oraz normy euklidesowej wektora dokumentu *j*. Tak przyjęto miara kosinusoidalna nie faworyzuje dokumentów o większej ilości termów, ponieważ kosinus kąta jest niewrażliwy na długość wektorów. Zastosowanie dzielenia przez opisany powyżej mianownik powoduje normalizacje długości wektorów q i d_j [15].



Rys. 6.1: Kosinusoidalna miara podobieństwa w przestrzeni R^3. [16]

Im mniejszy kąt pomiędzy wektorami (większy cosinus) tym większe podobieństwo dokumentów, analogicznie im większy kąt pomiędzy wektorami (mniejszy cosinus) tym mniejsze podobieństwo pomiędzy dokumentami.

Szczególnym przypadkiem potwierdzającym poprawność miary kosinusoidalnej jest porównanie dokumentu tekstowego d i dokumentu d', gdzie dokument d' posiada zdublowany tekst z dokumentu d. Pomimo podwójnej długości dokumentu *d'*, rozkład termów w dokumencie *d'* jest identyczny z rozkładem termów w dokumencie *d*. Z tego powodu podobieństwo pomiędzy dokumentami *d* i *d'* wyliczone z użyciem miary kosinusoidalnej jest dokładnie równe jeden.

Dzięki możliwości wyliczenia podobieństwa dla dowolnych dokumentów, model przestrzeni wektorowej pozwala na ocenę stopnia relewancji każdego dokumentu dla danego zapytania, a więc możliwym wynikiem zapytania jest utworzony ranking dokumentów, posortowany malejąco według stopnia relewancji. Najbardziej relewantne dokumenty prezentowane są użytkownikowi na początku rankingu, najmniej relewantne – na końcu. Ułatwia to przeglądanie wyników zapytania złożonych z wielu dokumentów.

W tab. 6.3 zaprezentowano przykładową macierz term-dokument wraz z zapytaniem *q*.

	*d*1	*d*2	*d*3	*q*
term a	8	3	1	9
term b	7	5	5	5
term c	4	1	3	6
term d	0	7	1	2

Tab. 6.3: Przykładowa macierz term-dokument wraz z zapytaniem *q*. Opracowanie własne.

Korzystając z miary kosinusoidalnej, możliwe jest obliczenie podobieństwa każdego dokumentu d_j dla zapytania *q*:

$$\mathrm{sim}(q,d_1)=\frac{8*9+7*5+4*6+0*2}{\sqrt{(8^2+7^2+4^2+0^2)*(9^2+5^2+6^2+2^2)}}=\frac{131}{\sqrt{129*146}}=95{,}46\,\%$$

$$\mathrm{sim}(q,d_2)=65{,}02\,\%$$

$$\mathrm{sim}(q,d_3)=74{,}48\,\%$$

Dzięki temu możliwe jest utworzenie rankingu dokumentów, posortowanego malejąco według stopnia relewancji:

*d*1
*d*3
*d*2

Tab. 6.4: Ranking dokumentów dla zapytania *q*. Opracowanie własne.

6.4. Problemy związane z modelem przestrzeni wektorowej

Model przestrzeni wektorowej posiada oczywiście swoje wady i zalety. Oczywistą zaletą modelu wektorowego w porównaniu do modelu boolowskiego jest możliwość rezygnacji z ograniczających warunków w postaci binarnej, prawdy lub fałszu. Umożliwia to wyliczenie bardziej precyzyjnej wartości podobieństwa, a zatem także pozwala na tworzenie rankingu podobieństwa dokumentów dla danego zapytania. Precyzyjne określenie miary podobieństwa pozwala zatem na zastosowanie logiki rozmytej – dokument może zostać zaklasyfikowany częściowo do ustalonej kategorii. Model przestrzeni wektorowej charakteryzuje się również wieloma innymi problemami, dlatego podjęto próby zrekompensowania części z poniższych wad, poprzez technikę *tf*idf* oraz zastosowanie tezaurusów, słowników synonimów i tzw. stop list. [15]

Problem synonimów
Problem synonimów oznacza, że dane słowo nie pojawia się w dokumencie, chociaż dokument jest ściśle związany z tym słowem, poprzez występowanie w nim jego synonimu np. "samochód" oraz "pojazd". Dokumenty o zbliżonym kontekście, lecz wykorzystujące różny zasób słownictwa, mogą zostać określone jako nierelewantne do siebie, co zakończy się fałszywie negatywną rekomendacją. Podczas wyszukiwania termów w danym dokumencie system powinien również uwzględnić synonimy danego termu.

Problem polisemii
Problem polisemii oznacza, że ten sam term może mieć różne znaczenia w różnych kontekstach, np. term "mining" oznacza wydobycie węgla lub eksploracje danych w kontekście informatycznym.

Problem długich dokumentów
Dokumenty o większej długości narażone są na niskie wartości iloczynu skalarnego wektora zapytania i wektora dokumentu, ponieważ ilość występujących termów w obu dokumentach jest zbyt duża.

Problem niezależności termów
Wadą modelu przestrzeni wektorowej jest oczywisty brak uwzględniania semantyki tekstu oraz kolejności występowania termów po sobie – uwzględniona jest tylko częstotliwość każdego termu. Również każdy term jest traktowany niezależnie i nie uwzględnia to faktu, że jedne termy mogą mieć większą wartość dyskryminacyjną niż inne.

6.5. Technika Tf-idf

Zwykła częstotliwość termów w modelu przestrzeni wektorowej posiada bardzo poważną wadę – wszystkie termy są równie istotne w procesie wyliczania stopnia relewancji. W rzeczywistości część termów posiada niewielką wartość dyskryminacyjną w procesie wyznaczania relewancji. Przykładowo, wysoce prawdopodobne jest, że w kolekcji dokumentów z branży motoryzacyjnej, większość dokumentów będzie posiadała term "auto". Z tego powodu wprowadzono między innymi technikę Tf-idf, która służy wyważeniu wartości częstotliwości termu (w zakresie lokalnym dokumentu) oraz częstotliwości występowania termu w całej kolekcji (zakres globalny) [16].

Wagi termów w modelu tf-idf uwzględniają dwa podstawowe elementy oceny istotności termu:

- Częstość termu *tf* (ang. term frequency)
  Liczba wystąpień termu w dokumencie lub inna miara znaczenia termu dla treści konkretnego dokumentu.
- Odwrotność częstości dokumentu *idf* (ang. inverse document frequency)
  Miara wartości informacyjnej termu, czyli jego przydatności dla rozróżniania treści różnych dokumentów w kolekcji.

Jeśli term występuje w wielu dokumentach, to jego wartość dyskryminacyjna jest relatywnie niższa. Może więc być to po prostu odwrotność liczby dokumentów, w których występuje dany term: 1/*df*(i). Zazwyczaj jednak stosuje się przyjętą na drodze empirycznej nieco zmodyfikowaną miarę *idf*(i) = log(*N*/*df*(i)), gdzie N jest liczbą dokumentów w kolekcji [16].

$$w_{ij} = tf_{ij} \cdot idf_i = tf_{ij} \cdot \log\left(\frac{N}{df_i}\right)$$

,gdzie:

tf_{ij} – częstość termu *i* w dokumencie *j*,

idf_i – odwrotność częstości dokumentów

N – liczba dokumentów w kolekcji

df_i – liczba dokumentów zawierających term *i*.

Wadą współczynnika tf-idf jest wykorzystywanie w obliczeniach łącznej liczby dokumentów zawierających dany term. Dlatego wagę tę należy obliczać każdorazowo przy każdej aktualizacji dokumentów.

6.6. Słowniki

Poza poprawieniem jakości wyszukiwania, stosowanie różnego rodzaju słowników pozwala na redukcję ilości termów, zwiększając także szybkość wyszukiwania. Korzystanie z słowników nie gwarantuje zachowanie lingwistycznego znaczenia i w głównej mierze jest zależne od rodzaju posiadanych danych tekstowych. Podczas rozpoznawania termu, każdy słownik w podanej kolejności jest kolejno odpytywany o zadany term, aż któryś z słowników udzieli pomyślnej odpowiedzi. W przypadku gdy żaden ze słowników nie potrafił zwrócić pomyślnej odpowiedzi lub gdy term został zidentyfikowany jako "stop word", term zostaje odrzucony i nie jest indeksowany przez system wyszukiwawczy. Generalną zasadą przy stosowaniu słowników jest używanie najpierw bardziej specjalistycznych słowników, a potem bardziej ogólnych, kończąc na tych, które potrafią zawsze udzielić pomyślnej odpowiedzi (np. algorytm Snowball) [17].

Stop listy (ang. Stop words)
Systemy do wyszukiwania informacji często wiążą ze zbiorem dokumentów tzw. „stop listę”, zawierającą zbiór słów powszechnie występujących często w dokumentach i nie posiadających przez to dobrej wartości dyskryminacyjnej. Najczęściej są to spójniki, przyimki, zaimki oraz inne słowa, które nie niosą samodzielnie żadnego znaczenia. Do typowych słów należących do stop listy w języku angielskim można zaliczyć wyrazy np."a, the, of, for, with, etc".

Konwersja termów do rdzenia
Konwersja termów do rdzenia oznacza sprowadzenie wyrazów do formy rdzeniowej. Różne formy tego samego wyrazu takie jak np. liczba mnoga, formy odczasownikowe powinny być sprowadzone do tej samej formy, aby nie zostały potraktowane jako dwa odmienne termy. Przykładowo, dla każdego ze słów "computer", "computing", "computation", "computer" rdzeniem jest "comput". W celu sprowadzenia wielu różnych form językowych do tej samej formy, używa się różnego rodzaju słowników morfologicznych takich jak Ispell Dictionary czy algorytmów jak Porter Stemming Algorithm [18]. W przypadku stosowania słowników morfologicznych mówi się o zjawisku lematyzacji (ang. Lemmatization). Zagadnienia te zostaną opisane szerzej w kolejnym podrozdziale.

Słownik nazw własnych
Słownik nazw własnych pozwala na zdefiniowanie termów, które nie powinny być sprowadzone do rdzenia. Przykładowo, bez słownika nazw własnych wyraz "Paris" byłby sklasyfikowany przez algorytm dokonujący konwersji do rdzenia jako forma mnoga, i odpowiednio zamieniony do termu "pari".

Słownik synonimów
Słownik synonimów pozwala na zastąpienie danego termu jego synonimem. Podstawową zaletą stosowania słownika synonimów jest możliwość uwzględnienia synonimów danego termu.

Tezaurusy
Tezaurus jest słownikiem synonimów, który podaje dla każdej pozycji słownika jeden lub więcej kategorii pojęć lub klas pojęć, co pozwala na zastąpienie kilku pojęć przez jedną klasę lub odwzorowanie pojęć niejednoznacznych w określoną kategorię.

6.7. Tworzenie reprezentacji wektorowej dokumentu

Tworzenie reprezentacji wektorowej dokumentu jest skomplikowanym procesem, korzystającym także z osiągnięć z dziedziny przetwarzania języka naturalnego (ang. natural language processing). Na początku zostaną przytoczone podstawowe definicje [15]:

Token – sekwencja znaków w konkretnym dokumencie, zgrupowana razem w jedną semantyczną jednostkę w celu dalszego przetwarzania przez system, przykładowo "Finance".

Tokenizacja – podział tekstu na tokeny, usuwający dodatkowo niepotrzebne znaki, np. znaki interpunkcyjne. Bardziej zaawansowane analizatory tekstu mogą ekstrahować również całe frazy.

Typ – klasa wszystkich tokenów zawierających te same sekwencje znaków.

Term – znormalizowany typ, które zostanie finalnie wykorzystany w procesie wyszukiwania informacji.

Normalizacja tokenów – proces modyfikacji tokenów w celu wydobycia jednej wspólnej formy, która będzie mogła później zostać użyta do spójnego sposobu indeksacji dokumentów jak i zapytań. Przykładowo, wpisując frazę "USA" należy również dopasować do niej frazę "U.S.A". Normalizacja tekstu zakłada również konwersję całego tekstu na małe lub wielkie litery. Konwersja tekstu (najczęściej na małe litery) powinna być wykonana pomimo tego, że słowa o różnej wielkości mogą mieć różne znaczenie, przykładowo: "MIT" (Massachusetts Institute of Technology) oraz "mit"(niemiecki przyimek).

Zadanie wykonania poprawnej tokenizacji stawia przed wieloma wyzwaniami. Przykładowo, czy poniższe frazy powinny zostać podzielone na dwa osobne tokeny czy powinny być uwzględnione jako jedna całość?

- Hewlett-Packard
- data base
- San Francisco
- co-education

Tokenizacja jest również problematyczna w wielu innych językach, przykładowym problemem są złożenia niemieckich słów np. Lebensversicherungsgesellschaftsangestellter (pracownik firmy ubezpieczeń na życie). Rozwiązaniem tych przypadków w języku niemieckim są moduły rozdzielania słów złożonych (ang. compound-splitter module), które próbują sprawdzić czy dane złożone słowo może zostać podzielone na mniejsze słowa występujące w słowniku. Niestety technika ta osiągnęła swój limit w językach pochodzących z Azji Wschodniej (Chiński, Japoński, Koreański), gdzie tekst jest pisany bez żadnych spacji pomiędzy słowami. Innym przykładem jest ambiwalentność znaczenia niektórych chińskich sekwencji znaków, przykładowo sekwencja:

和尚

oznacza słowo "mnich" gdy zostanie potraktowana jako jedna całość lub oznacza odpowiednio słowa "i" oraz "wciąż" gdy zostanie rozdzielona na dwie części.

Również niezbędne jest rozwiązaniem problemów ze znakami diakrytycznymi (szczególnie w języku polskim) oraz akcentami. Przykładowo, francuskie słowo "résumé" zostaje znormalizowane do słowa "resume″, a niemieckie słowo "universität" posiadające umlaut zostaje znormalizowane do słowa "universitaet".

Z przyczyn gramatycznych, dokumenty posiadają różne formy podstawowego słowa jak np. "democracy", "democratic", "democratization". Należy je sprowadzić do tej samej formy (rdzenia), aby wyszukując informacje z użyciem jednego z powyższych słów, zwrócić dokumenty posiadające również inne formy danego słowa. Celem zarówno lematyzacji jak i stemmingu, jest normalizacja różnych form danego słowa do jednej wspólnej podstawowej formy słowa. Przykładowo:
formy "am", "are", "is" powinny zostać sprowadzone do formy podstawowej "be", a formy "car", "cars", "car's" powinny zostać sprowadzone do formy podstawowej "car".

Chociaż zarówno celem lematyzacji oraz stemmingu jest zwiększenie skuteczności algorytmu, jednak oba algorytmy działają zupełnie odmiennie. Stemming jest surowym heurystycznym procesem usuwania końcówek słów. Przykładowo, pierwszym krokiem w popularnym algorytmie stemmingu Martina Portera jest dokonanie konwersji na podstawie następujących reguł usuwania końcówek słów [18]:
"SSES" -> "SS"
"IES" -> "I"
"S" -> ""

Z kolei lematyzacja odnosi się do analizy słów z użyciem morfologicznych słowników w celu sprowadzenia danej formy słowa do rzeczywistej podstawowej formy danego słowa w danym języku. Podstawową formę danego słowa określa się lemmem (ang. lemma) [15].

W tab. 6.5 zaprezentowano przykładowe porównanie algorytmu stemmingu Snowball (bazującego na algorytmie stemmingu Portera) z algorytmem lematyzacji działającego w oparciu o słownik morfologiczny języka angielskiego Ispell:

	Stemming (Snowball)	**Lematyzacja (Ispell)**
automate	autom	automate
automation	autom	automate
cat	cat	cat
cats	cat	cat
combinator	combin	combinator
combinatoric	combinator	combinatoric
committee	committe	committee
committees	committe	committee
complicate	complic	complicate
complicated	complic	complicated | complicate
confidence	confid	confidence
create	creat	create
creative	creativ	creative | create
creativity	creativ	creativity
people	peopl	people
ponies	poni	pony
pony	poni	pony

Tab. 6.5: Porównanie algorytmu stemmingu Snowball z algorytmem lematyzacji w oparciu o słownik morfologiczny języka angielskiego Ispell. Opracowanie własne.

Oczywistym wnioskiem jest, że stemming powoduje utworzenie błędnych skróconych form słów, lecz dla systemu wyszukiwawczego nie ma to żadnego znaczenia, ponieważ głównym celem dla algorytmów wyszukiwania jest dopasowanie termów występujących w dokumentach do termów występujących w zapytaniu. Dla tokenów "automate" i "automation" oraz "ponies" i "pony" oba algorytmy zwróciły identyczny znormalizowany token odpowiedni dla swojego algorytmu. W przypadku tokenów "cat","cats" i "committee","committees" oba algorytmy zwróciły ten sam token. Słownik Ispell poradził sobie z dopasowaniem tokenów "create" oraz "creative" (dzięki zwróconej alternatywie tokenów), z kolei algorytm Snowball poradził sobie z dopasowaniem tokenów "creative" oraz "creativity".

Według [15], normalizacja z użyciem lematyzacji nie poprawia znacząco skuteczności wyszukiwania informacji w języku angielskim w stosunku do normalizacji z użyciem stemmingu, oraz niestety pogarsza też szybkość wyszukiwania. Autorzy twierdzą, że stemming powiększa wartości zwrotu a pogarsza wartości precyzji (opisane w podrozdziale 7.4). Temat ten zostanie poruszony szerzej w przeprowadzonych badaniach.

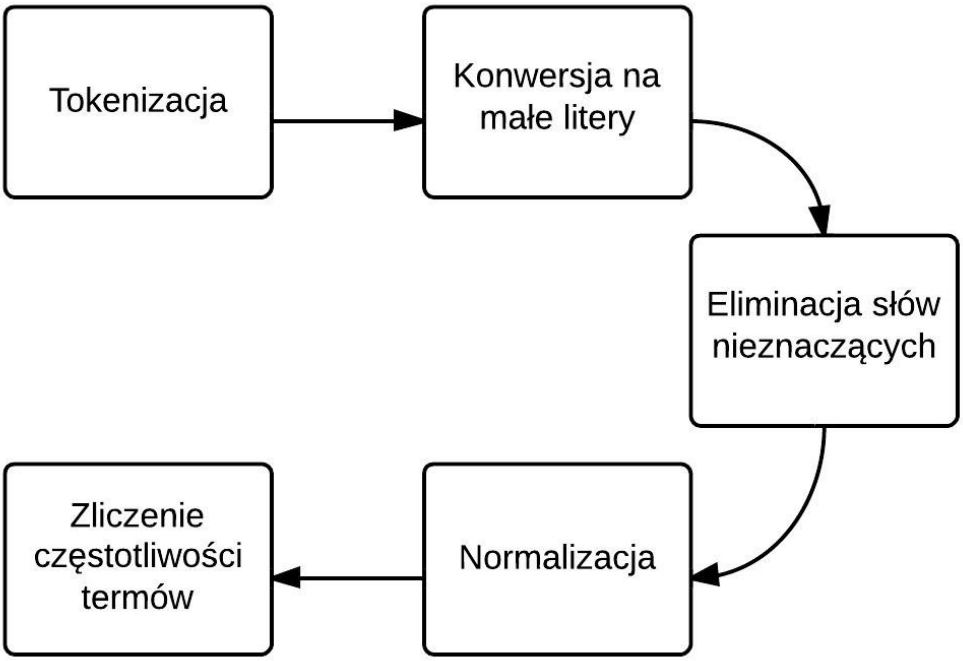
W tab. 6.6 przedstawiono utworzoną reprezentację wektorową dla przykładowego dokumentu o tematyce finansowej. Tekst został wpierw podzielony na tokeny (usunięto m. in. znaki interpunkcyjne). Następnie dokonano konwersji wszystkich znaków do małych liter oraz usunięto wszystkie słowa nieznaczące ("a, an, and, are, for, of, that, they, their, to"). Dzięki lematyzacji w oparciu o słownik morfologiczny języka angielskiego Ispell, utworzono finalnie końcowe termy określające dany dokument tekstowy. Następnie zliczono częstotliwość każdego termu.

Economists make a number of abstract assumptions for purposes of their analyses and predictions. They generally regard financial markets that function for the financial system as an efficient mechanism (Efficient-market hypothesis). Instead, financial markets are subject to human error and emotion.

abstract	1
analyses	1
assumption	1
economist	1
efficient	2
emotion	1
error	1
financial	3
function	1
general	1
human	1
hypothesis	1
instead	1
make	1
market	3
mechanism	1
number	1
prediction	1
purpose	1
regard	1
subject	1
system	1

Tab. 6.6: Reprezentacja wektorowa dokumentu tekstowego po dokonaniu tokenizacji, normalizacji oraz usunięciu słów nieznaczących z wykorzystaniem słownika Ispell. Opracowanie własne.

Podsumowując, na rys. 6.2 zaprezentowano pełny schemat tworzenia reprezentacji wektorowej dokumentu tekstowego.



Rys. 6.2: Schemat tworzenia reprezentacji wektorowej dokumentu. Opracowanie własne.

6.8. Inne techniki wyszukiwania informacji

Jedną z kolejnych technik wyszukiwania informacji jest rozszerzony model boolowski (ang. extended boolean model). Celem rozszerzonego modelu boolowskiego jest rozwiązanie problemów wynikających z podstawowego modelu boolowskiego. Standardowy model boolowski nie uwzględnia wag odpowiednich termów w zapytaniu, co powoduje ze zbiór wyników w odpowiedzi na wyszukiwanie jest bardzo często albo za duży albo za mały. Celem rozszerzonego modelu boolowskiego jest wprowadzenie możliwości częściowego dopasowania do zapytania oraz ważenia wag termów. Z tego powodu rozszerzony model boolowski może być uznany jako uogólnienie zarówno podstawowego modelu boolowskiego oraz modelu przestrzeni wektorowej [19].

Alternatywną techniką wyszukiwania informacji jest model probabilistyczny, bazujący na statystyce Bayesowskiej. Kosinusoidalna miara podobieństwa w modelu przestrzeni wektorowej dokładnie nie oddaje czy dany dokument jest odpowiedni dla zapytania użytkownika [15]. Przykładowo, dokument posiadający wysoką wartość podobieństwa (wyliczoną na podstawie kosinusoidalnej miary podobieństwa), może być wysoko relewantny dla zapytania, lub także całkowicie nierelewantny. Dlatego podstawą teoretyczną modeli probabilistycznych jest tzw. zasada rankingowania probabilistycznego (ang. probability ranking principle). Zgodnie z nią optymalne działanie systemu wyszukiwawczego może zostać osiągnięte poprzez rankingowanie dokumentów zgodnie z prawdopodobieństwem ich oceny jako relewantnych dla zapytania.[16]. Pełna definicja zasady rankingowania probabilistycznego brzmi:

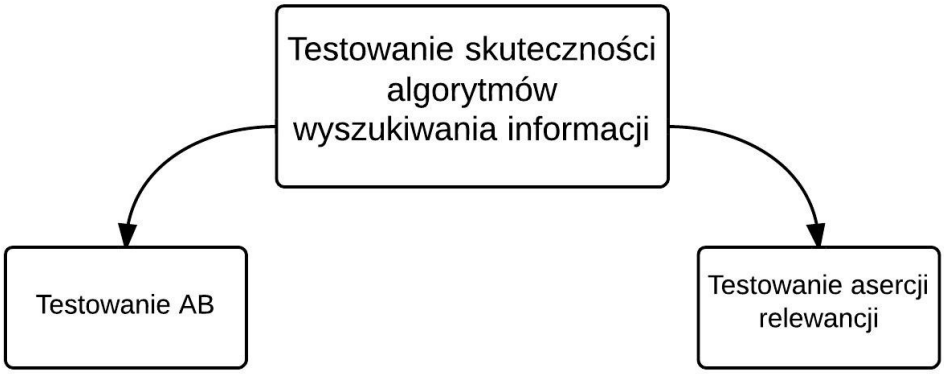
„Jeżeli odpowiedzią systemu na każde zapytanie użytkownika jest ranking dokumentów w kolejności malejącej zgodnie z prawdopodobieństwem ich oceny jako relewantnych dla zapytania, gdzie prawdopodobieństwa są estymowane jak najdokładniej jak to możliwe na podstawie wszystkich możliwych danych dostępnych do tego celu, to wtedy ogólna skuteczność systemu w wyszukiwaniu informacji dla danego użytkownika będzie najlepszą z możliwych jaka może być osiągnięta na podstawie podanych danych." [15].

7. Testowanie skuteczności algorytmów wyszukiwania informacji

Ocena jakości algorytmów wyszukiwania informacji relewantnych jest kluczowym czynnikiem w projektowaniu, implementacji oraz zarządzaniu efektywnymi systemami wyszukiwania informacji, ponieważ pozwala ona na zmierzenie skuteczności zaspokajania potrzeb informacyjnych użytkowników. Należy postawić pytanie, czym charakteryzuje się skuteczny system do wyszukiwania informacji relewantnych. Skuteczność ta może być różnie postrzegana, przykładowo skuteczność może odnosić się do zdolności rozróżniania informacji relewantnych lub nierelewantnych. Innym kryterium może być szybkość z jaką system potrafi zwrócić odpowiedzi na zapytania lub satysfakcja użytkowników z uzyskanych odpowiedzi. Kolejnym kryterium może być łatwość obsługi systemu przez użytkownika. Przez ostatnie 50 lat ewaluacja skuteczności systemów wyszukiwania informacji była przedmiotem wielu badań, dyskusji i kontrowersji.

Algorytm do wyszukiwania informacji relewantnych powinien charakteryzować się dużą skutecznością w wyznaczaniu wyników, w przeciwnym przypadku przestał by być użyteczny dla użytkownika. Przykładowo, głównym czynnikiem, któremu wyszukiwarka „Google" zawdzięcza swoją popularność jest fakt, że wyszukiwarka ta jest wyjątkowo szybka i skuteczna, co oznacza, że w większości przypadków, wyszukiwarka rzeczywiście prezentuje w odpowiedniej kolejności najbardziej relewantne strony dla danego zapytania.

Aby wyszukiwarka mogła podać odpowiedź na podane zapytanie, musi posiadać wiedzę na temat tego, jak dana informacja jest relewantna dla danego zapytania. Na podstawie tego można zdefiniować i rozdzielić dwa różne sposoby testowania skuteczności algorytmów wyszukiwania informacji relewantnych (rys. 7.1).



Rys. 7.1: Klasyfikacja metod testowania skuteczności algorytmów wyszukiwania informacji. Opracowanie własne.

7.1. Testowanie AB

Testowanie AB (ang. AB Testing) jest ogólną techniką wykorzystywaną w trakcie sprawdzania nowych funkcjonalności. Taki sposób testowania zwany jest również testowaniem preferencji użytkownika. Przykładowo firma Google, chcąc wprowadzić nową funkcjonalność, przekierowuje drobną część swoich użytkowników na strony ze zmodyfikowanym interfejsem. Następnie na podstawie zachowania tej części internautów którzy korzystają z nowego interfejsu, firma jest w stanie zanalizować jak nowe zmiany wpłynęły by na zadowolenie oraz zachowanie wszystkich pozostałych użytkowników. [20]

W dziedzinie wyszukiwania informacji, testowanie AB jest bardzo prostym sposobem testowania skuteczności algorytmów wyszukiwania informacji relewantnych. Charakteryzuje się on brakiem wiedzy a priori odnośnie idealnych wyników zapytań o relewantność zadanych informacji. Każdy algorytm zwraca odpowiedź na zadane przez użytkownika zapytanie, a użytkownik wybiera rozwiązanie któregoś z algorytmów. Alternatywą jest również możliwość oceny przez testera podanych rozwiązań, bądź ich uszeregowanie w kolejności od najlepszego rozwiązania do najgorszego.

Cechy charakterystyczne:

- Rejestrowanie każdego wyboru użytkownika
- Brak wiedzy a priori
- Łatwy start oraz sposób konstrukcji testu
- Wysoki czynnik niepewności testu z powodu subiektywnych ocen użytkownika

7.2. Testowanie asercji relewancji

Głównym założeniem testowania asercji relewancji (ang. Relevancy Assertion Testing) jest fakt, że odpowiedzi na zapytania systemu o relewantność informacji, są z góry określone (wiedza a priori). Innym określeniem tego typu testowania jest również „Testowanie z absolutną prawdą”. Każde zapytanie do systemu o relewantność informacji zostało wcześniej wykonane i poddane ekspertyzie. Ekspert dziedzinowy lub system zdolny do podania rozwiązania, podał odpowiedź na każde zapytanie, które będzie wykorzystane w tego typu testowaniu, przykładowo: „jak podana informacja jest relewantna dla danego zapytania”.

Cechy charakterystyczne [15]:

- Ustalony zbiór informacji
- Ustalony zbiór zapytań odnośnie relewancji informacji należących do powyższego zbioru informacji
- Ustalony zbiór idealnych wyników zapytań o relewancję zadanych informacji
- Wykorzystanie kilku różnych algorytmów (silników) wyszukiwania informacji relewantnych dla danego zapytania
- Wyniki każdego algorytmu są zgrupowane razem dla każdego zapytania
- Wyniki zapytań dla różnych algorytmów mogą być ze sobą porównywane, ponieważ istnieje zbiór idealnych wyników zapytań.
- Wykorzystywane jest wiele rożnych formuł z dziedziny statystyki matematycznej, pozwalających ocenić przydatność każdego z algorytmów, na podstawie analizy wyników

7.3. Metodyka Cranfield

Jedną z najbardziej popularnych metod testowania skuteczności algorytmów wyszukiwania informacji jest ewaluacja oceny w oparciu o kolekcję dokumentów. Metodyka ta, bazująca na testowaniu asercji relewancji, pozwala na przeprowadzenie w pełni powtarzalnych serii eksperymentów, pozwalających na porównywanie różnych strategii wyszukiwania informacji [15]. Metodyka ta jest często określana jako metodyka Cranfield, której pierwsze serie eksperymentów zostały wykonane w Wielkiej Brytanii już w latach 1958-1966

Metodyka Cranfield składa się następujących po sobie operacji:

1. Przygotowanie różnych strategii wyszukiwania informacji, w celu ich późniejszego porównania.
2. Wykonanie serii testów (przebiegów) dla każdego z zapytań (tematów) dla każdej strategii w celu stworzenia rankingu dokumentów.
3. Wyliczenie skuteczności każdego zapytania dla każdej strategii, przy użyciu funkcji oceny algorytmów.
4. Wyliczenie średniej skuteczności ze wszystkich zapytań dla każdej strategii
5. Użycie końcowych rezultatów w celu porównania bądź stworzenia rankingu strategii wyszukiwania informacji.
6. Przeprowadzenie dodatkowych badań statystycznych w celu stwierdzenia czy różnice w skuteczności różnych strategii wyszukiwania informacji są rzeczywiście znaczące.

Metodyka Cranfield jest bardzo popularną metodą pomagającą twórcom w implementacji różnych strategii wyszukiwania informacji. Powstało wiele środowisk testujących, pozwalającym wielu pasjonatom i naukowcom na standaryzowaną ewaluacje jakości algorytmów wyszukiwania informacji z użyciem identycznej konfiguracji środowiska. Stabilność oraz standaryzacja tego rodzaju metodyki przyczyniły się w dużej mierze do jej popularności, w przeciwieństwie do manualnych testów użytkowych (testowanie AB), które choć są bardzo korzystne, to jednak są kosztowne i trudne do powtórzenia w celu uzyskania obiektywnych wyników. Metoda Cranfield, pomimo oczywistej swojej atrakcyjności, posiada wiele ograniczeń, ponieważ w dużej mierze jej abstrakcyjność odbiega od rzeczywistości, w której pracują systemy wyszukiwania informacji. Testowanie w oparciu o kolekcję dokumentów w metodyce Cranfield niestety opiera się na wielu założeniach [15]:

- Relewantność dokumentów jest niezależna od siebie
- Każdy dokument jest równo ważny dla użytkownika
- Potrzeba informacyjna użytkownika pozostaje cały czas stała
- Ustalony zbiór idealnych wyników dla zapytań jest reprezentatywny dla całej populacji

7.4. Funkcje oceny algorytmów

Testowanie asercji relewancji wiąże się z użyciem wielu funkcji matematycznych wyznaczających istotne cechy skuteczności zadanego algorytmu. Poniższe funkcje będą rozpatrywane w kontekście rekomendacji zaproponowanej użytkownikowi przez system, czyli informacji relewantnej dla jego zapytania. Funkcje te przybierają wartości od zero do jeden, i powinny dążyć do jak największej wartości. Poniższe definicje zostały opracowane na podstawie [11].

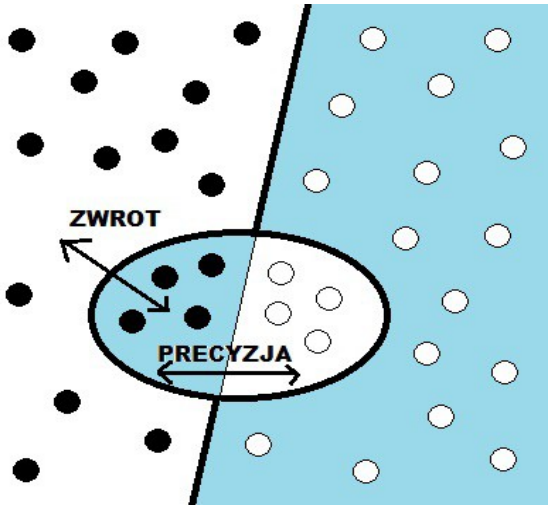
Precyzja (ang. precision) – iloraz sumy informacji, które zostały wyszukane i są rzeczywiście relewantne z punktu widzenia zapytania, do sumy wszystkich wyszukanych informacji dla danego zapytania.

$$Precyzja = \frac{|Informacje\ relewantne \cap Informacje\ wyszukane|}{|Informacje\ wyszukane|} = \frac{Rekomendacje\ prawdziwie\ pozytywne}{Rekomendacje\ prawdziwie\ pozytywne + Rekomendacje\ fałszywie\ pozytywne}$$

Zwrot (ang. recall) – iloraz sumy informacji, które zostały wyszukane i są rzeczywiście relewantne z punktu widzenia zapytania, do sumy wszystkich rzeczywiście relewantnych informacji dla danego zapytania. Miara zwrotu jest także często nazywaną miarą wrażliwości, ponieważ odpowiada ona zdolności algorytmu do identyfikacji rekomendacji pozytywnych.

$$Zwrot = \frac{|Informacje\ relewantne \cap Informacje\ wyszukane|}{|Informacje\ relewantne|} = \frac{Rekomendacje\ prawdziwie\ pozytywne}{Rekomendacje\ prawdziwie\ pozytywne + Rekomendacje\ fałszywie\ negatywne}$$

Na rys. 7.2 zaprezentowane dwie pierwsze miary, zwrot oraz precyzję:



Rys. 7.2: Prezentacja miary zwrotu i miary precyzji. Po lewej stronie zaprezentowano informacje relewantne, po prawej nierelewantne. Okrąg zawiera informacje wyszukane. Niebieski region reprezentuje rekomendacje prawdziwe. Opracowanie własne.

Funkcje precyzji i zwrotu konkurują ze sobą, tzn. że najczęściej wzrost jednej wartości z nich powoduje zmniejszenie drugiej wartości. Dobry algorytm powinien dążyć do osiągnięcia jednocześnie dużej precyzji i dużego zwrotu. Obie miary są równie istotne, dlatego aby uwzględnić ten fakt, stworzoną dodatkową F-miarę, będącą średnią harmoniczną tych dwóch miar.

F-miara (ang. F-measure) – średnia harmoniczna precyzji i zwrotu.

$$F = \frac{2 * zwrot * precyzja}{zwrot + precyzja}$$

Skuteczność (ang. accuracy) – iloraz sumy informacji, które zostały wyszukane i są rzeczywiście relewantne oraz informacji, które nie zostały wyszukane i nie były rzeczywiście relewantne z punktu widzenia zapytania, do sumy wszystkich informacji.

$$Skuteczność = \frac{|(Informacje\,relewantne \cap Informacje\,wyszukane) \cup (Informacje\,nierelewantne \setminus Informacje\,wyszukane)|}{|Wszystkie\,informacje|} =$$

$$= \frac{Rekomendacje\ prawdziwie\ pozytywne + Rekomendacje\ prawdziwie\ negatywne}{Wszystkie\,rekomendacje}$$

Niestety miara skuteczności nie jest odpowiednią miarą dla problemów wyszukiwania i rekomendacji informacji. W większości przypadków, informacje są bardzo zniekształcone, co powoduje, że ogromna liczba dokumentów należy do kategorii informacji nierelewantnych. System rekomendacyjny nastawiony na maksymalizację skuteczności, może dążyć do uznawania jak największej ilości dokumentów za nierelewantne dla dowolnych zapytań. Klasyfikowanie większości dokumentów jako nierelewantne wprawdzie maksymalizuje skuteczność i chroni przed rekomendacjami fałszywie pozytywnymi, jednak jest w zupełności nie przydatne dla użytkownika. Użytkownik zawsze woli zobaczyć kilka dokumentów w odpowiedzi na dane zapytanie, mając pewną tolerancję na wystąpienie fałszywie pozytywnych rekomendacji. Z tego powodu dużą większą wartość dla problemów rekomendacyjnych mają funkcje precyzji i zwrotu, które koncentrują się w głównej mierze na rekomendacjach prawdziwie pozytywnych.

R – Precyzja (ang. R-Precision) – precyzja przy założeniu, że algorytm posiada wiedzę, jak wielki ilościowo zbiór informacji relewantnych powinien zaproponować. W tym przypadku zachodzi zależność:

$$|Informacje\,relewantne| = |Informacje\,wyszukane|$$

ponieważ podana liczebność zbioru informacji relewantnych do zaproponowania jest równa rzeczywistej liczności zbioru informacji relewantnych.

Dla tego przypadku, zachodzi zależność:

$$Precyzja = \frac{|Informacje\,relewantne \cap Informacje\,wyszukane|}{|Informacje\,wyszukane|} = \frac{|Informacje\,relewantne \cap Informacje\,wyszukane|}{|Informacje\,relewantne|} = Zwrot$$

F-miara jest średnią harmoniczną precyzji i zwrotu, a więc jest także w tym przypadku im równa:

$$Precyzja = Zwrot = F_{miara}$$

8. Klasyfikacja dokumentów tekstowych

8.1. Systematyka klasyfikacji dokumentów tekstowych

Klasyfikacja dokumentów polega na przypisaniu każdemu dokumentowi d_j z danego zbioru dokumentów, jednej ze skończonego zbioru ustalonych z góry kategorii (klas) [16]. Klasyfikacja dokumentów tekstowych jest szeroko stosowana w różnych dziedzinach przetwarzania tekstu, przykładowymi zastosowaniami klasyfikacji dokumentów tekstowych jest filtrowaniu spamu, automatyczne klasyfikowanie przychodzących wiadomości mailowych do odpowiednich folderów tematycznych, czy ochrona dzieci przed treściami pornograficznymi.

Poniżej została zaprezentowana podstawowa systematyka klasyfikacji [11]:

Podział ze względu na automatyzację:

- **Klasyfikacja manualna** – klasyfikacja została dokonana przez człowieka.
- **Klasyfikacja automatyczna** – klasyfikacja została dokonana z użyciem odpowiedniego algorytmu klasyfikacyjnego.

Podział klasyfikacji automatycznej ze względu na kontrolę poprawności klasyfikacji:

- **Klasyfikacja z nadzorem** - istnieje zewnętrzne źródło (najczęściej człowiek) potwierdzające poprawność klasyfikacji,
- **Klasyfikacja bez nadzoru** - nie ma możliwości dokładnego sprawdzenia poprawności klasyfikacji.

Podział klasyfikacji ze względu na ilość klas:

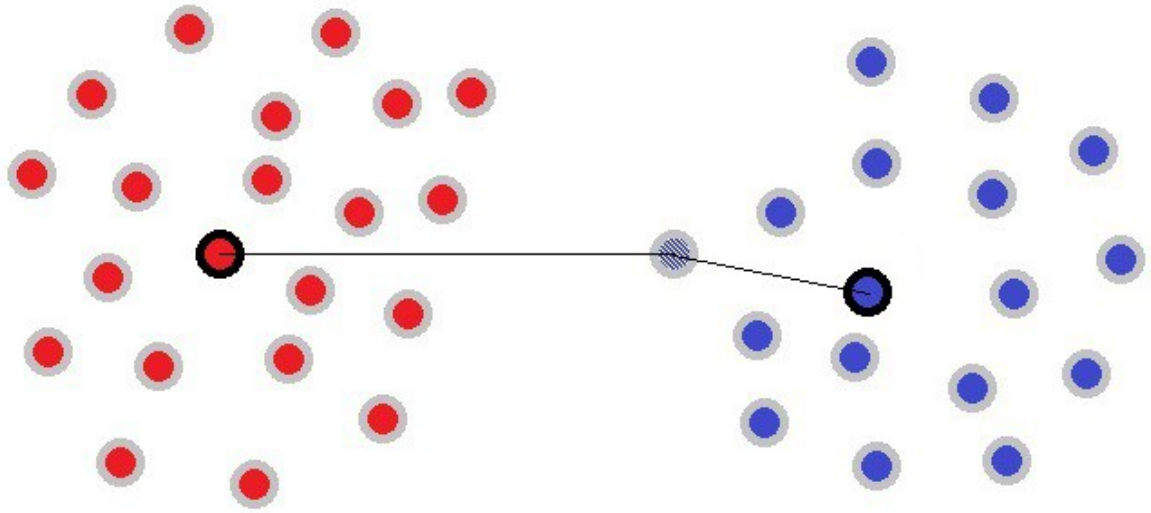
- **Klasyfikacja wieloklasowa** - istnieje przynajmniej 3 lub więcej klas do której może być zaklasyfikowany dokument
- **Klasyfikacja binarna** – dokument może być zaklasyfikowany tylko do jednej z dwóch klas. Przykładowo, filtrowanie spamu polega na zaklasyfikowaniu dokumentu wiadomości e-mail jako spamu lub jako wiadomości użytecznej dla użytkownika. Kolejnym przykładem jest klasyfikowanie dokumentów tekstowych uwzględniając tylko dwa możliwe kategorie: relewantne lub nierelewantne.

System do zarządzania informacją osobistą powinien umożliwiać użytkownikowi wykonanie manualnej klasyfikacji dokumentu do dowolnej kategorii jak i sugerować użytkownikowi wykonanie klasyfikacji danego dokumentu do określonej kategorii.

8.2. Klasyfikacja z użyciem profilów metodą Rocchio

Klasyfikacja dokumentów do danej klasy (kategorii) odbywa się na założeniu hipotezy bezpośredniego sąsiedztwa (ang. contiguity hypothesis) [15]. Hipoteza bezpośredniego sąsiedztwa polega na założeniu, że dokumenty należące do tej samej klasy (kategorii) są w bliskim sąsiedztwie, a dokumenty należące do różnych klas (kategorii) nie nachodzą na siebie. Przenieśmy klasyfikację dokumentów tekstowych na wektory przestrzeni N-wymiarowej przechowujących odpowiednie termy dla danych dokumentów. Przykładowo, dokumenty zaklasyfikowane do kategorii "Wielka Brytania" będą posiadały duże wartości wag "Londyn" oraz "Brytyjski" w porównaniu do dokumentów z kategorii "Chiny" posiadających duże wartości wag "Pekin" oraz "Chiński".

Klasyfikacja nowego dokumentu odbywa się na podstawie jego odległości do profili kategorii. Dokumentowi przypisywana jest kategoria, której profil jest najbardziej podobny do reprezentującego go wektora. Odległość pomiędzy centroidem a klasyfikowanym dokumentem w modelu przestrzeni wektorowej jest wyliczana najczęściej z użyciem kosinusoidalnej miary podobieństwa.



Rys. 8.1: Klasyfikacja dokumentu do odpowiedniej kategorii w przestrzeni R^2 na podstawie odległości od centroidów. Nowy dokument został zaklasyfikowany do kategorii niebieskiej z powodu najbliższej odległości do centroidu kategorii niebieskiej. Opracowanie własne.

Aby odpowiednio sklasyfikować dany dokument do określonej klasy (kategorii), niezbędne jest posiadanie profilu danej klasy (kategorii). Profil danej klasy (zwany również prototypem lub centroidem), zawiera typowe, charakterystyczne cechy odróżniające daną klasę (kategorię) od innych.

Poniżej zaprezentowano podstawowy podział sposobów tworzenia profilów klas (kategorii):

- Ręczne wybranie profilu
- Wyliczenie profilu na podstawie zbioru przykładowych dokumentów należących do danej klasy (kategorii) z użyciem algorytmu Rocchio

Algorytm Rocchio polega na utworzeniu centroidu (wektora średnich) dla każdej kategorii z wektorów wszystkich dokumentów należących do danej kategorii. Następnie rozpatrywany dokument jest klasyfikowany do klasy (kategorii) najbliższego centroidu [15].

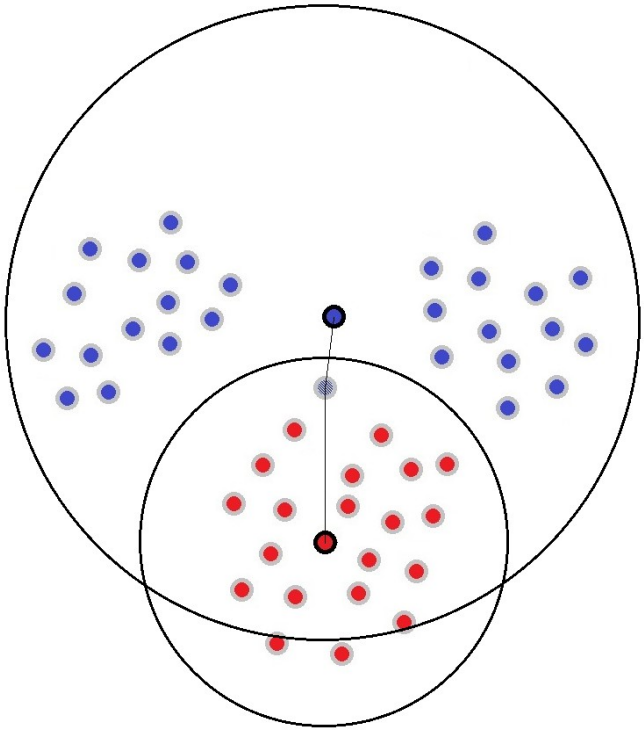
Centroid (wektor średnich) kategorii *c* jest utworzony z użyciem następującego wzoru:

$$\vec{\mu}(c) = \frac{1}{|D_c|} \sum_{d \in D_c} \vec{v}(d)$$

gdzie D_c jest zbiorem wszystkich dokumentów należących do kategorii *c* oraz

$\vec{v}(d)$ oznacza wektor dokumentu *d*.

Wadą algorytmu Rocchio jest słaba skuteczność klasyfikacji w pracy z kategoriami polimorficznymi (dysjunkcyjnymi) [16], co zaprezentowano na rys. 8.2.



Rys. 8.2: Klasyfikacja dokumentu do odpowiedniej kategorii w przestrzeni R^2 na podstawie odległości od centroidów. Nowy dokument został zaklasyfikowany błędnie do kategorii niebieskiej z powodu najbliższej odległości do centroidu kategorii niebieskiej. Opracowanie na podstawie [15].

W przypadku kategorii polimorficznych, dużo lepiej radzi sobie algorytm K najbliższych sąsiadów [16]. Alternatywnym rozwiązaniem dla powyższego przypadku jest podzielenie kategorii niebieskiej na 2 rozdzielne kategorie.

9. Sprzężenie zwrotne

Nawet najlepszy system do wyszukiwania informacji ma swój ograniczony zwrot. Użytkownicy mogą uzyskać parę relewantnych dokumentów w odpowiedzi na swoje zapytania, ale prawie nigdy nie otrzymają wszystkich relewantnych dokumentów. W wielu przypadkach nie będzie to potrzebne, ale istnieją przypadki w których zapotrzebowanie na wysoki zwrot jest krytyczne.

Jednym ze sposobów polepszenia zwrotu (oraz także precyzji) jest skorzystanie z techniki sprzężenia zwrotnego. [15] Sprzężenie zwrotne polega na zaangażowaniu użytkownika w proces wyszukiwania informacji. Użytkownik udziela feedbacku na temat relewancji zwracanych przez system dokumentów. Podstawowa procedura polega na wykonaniu następujących operacji:
- użytkownik dokonuje prostego zapytania
- system zwraca zbiór wyników
- użytkownik oznacza kilka dokumentów jako relewantne oraz nierelewantne.
- system uwzględnia feedback użytkownika w celu zwrócenia dokumentów odpowiadających bardziej potrzebie informacyjnej użytkownika.

Według [15], technika ta osiąga bardzo dobre rezultaty. Proces sprzężenia zwrotnego może również występować w następujących po sobie iteracjach, jednak według autorów, zazwyczaj jedna iteracja jest wystarczająca.

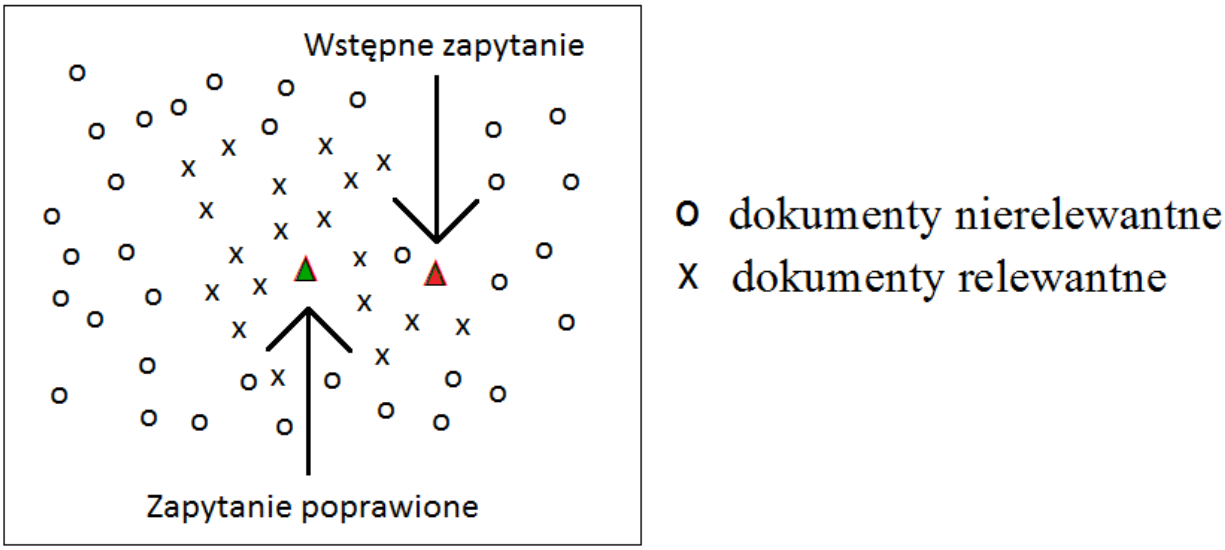
Idea sprzężenia zwrotnego w modelu przestrzeni wektorowej raz pierwszy została wprowadzona w systemie SMART w roku 1970. Opiera się ona na znalezieniu wektora q_{opt}, które zmaksymalizuje podobieństwo do dokumentów relewantnych i zminimalizuje podobieństwo do dokumentów nierelewantnych.

$$\vec{q}_{opt} = \arg\max_{\vec{q}}[\mathrm{sim}(\vec{q}, C_r) - \mathrm{sim}(\vec{q}, C_{nr})]$$

W modelu przestrzeni wektorowej, zapytanie te można opisać następująco [15]:

$$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$

, gdzie:
α – waga pierwotnego zapytania
β – waga dla dokumentów relewantnych
γ – waga dla dokumentów nierelewantnych
D_r – zbiór dokumentów relewantnych
D_{nr} – zbiór dokumentów nierelewantnych
$\vec{q}_0$ – wektor pierwotnego zapytania
$\vec{q}_m$ – wektor zmodyfikowany
$\vec{d}_j$ – wektor dokumentu *j*



Rys. 9.1: Algorytm Rocchio. Modyfikacja wstępnego zapytania. Opracowanie własne.

Zapytanie te pozwala przesunąć zmodyfikowany wektor bliżej dokumentów relewantnych, jednocześnie oddalając je od dokumentów relewantnych. Finalny zmodyfikowany wektor jest następnie używany w sposób standardowy do wyszukiwania dokumentów w modelu przestrzeni wektorowej. Wagi α, β i γ pozwalają na zdefiniowanie zapytania w taki sposób, aby zachować balans pomiędzy wykorzystaniem ocenionych dokumentów przez użytkownika, a pierwotnym zapytaniem. Według [15], pozytywny feedback okazuje się dużo bardziej cenny niż negatywny. Z tego powodu współczynnik γ powinien być mniejszy od β. W wielu systemach tylko pozytywny feedback jest brany pod uwagę, z tego powodu współczynnik γ jest równy zero.

Pseudosprzężenie zwrotne (ang. Pseudo relevance feedback), pozwala na lokalną analizę zapytania bez ingerencji użytkownika. Metoda ta polega na standardowym wyszukaniu dokumentów relewantnych (dowolną techniką), a następnie na założeniu, że K pierwszych dokumentów jest rzeczywiście relewantne [16]. Oznaczone dokumenty jako relewantne są wykorzystane dalej w standardowym procesie sprzężenia zwrotnego, tak jakby rzeczywiście były wybrane przez użytkownika.

Podstawowym sposobem oceny poprawy wyników z użyciem algorytmu sprzężenia zwrotnego jest porównanie osiąganych wartości precyzji i zwrotu. Przykładowo, stosując funkcję R-precyzji, wystarczy obliczyć wartości dla pierwotnego zapytania i porównać ją z wartościami dla zmodyfikowanego zapytania po skorzystaniu z sprzężenia zwrotnego. Jednak ta metoda wskazywała by na zbyt wygórowaną poprawę wyników, ponieważ końcowa wartość R-precyzji uwzględniała by również dokumenty już ocenione przez użytkownika. Dlatego aby pozostać obiektywnym, należy w ocenie poprawy jakości wyników pominąć te dokumenty, które zostały wybrane przez użytkownika.

W 1969 roku Eleanor Ide opublikowała serię eksperymentów na bazie algorytmu Rocchio. [21] Na podstawie tych prac powstały trzy najbardziej skuteczne strategie na bazie algorytmu Rocchio. Pierwsza strategia jest bardzo podobna do oryginalnej metody, lecz pomija normalizację ilości dokumentów relewantnych i nierelewantnych:

$$\vec{q}_m = \alpha\vec{q}_0 + \beta \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$

Druga strategia jest podobna do pierwszej, lecz pozwala tylko na feedback z dokumentów relewantnych, co oznacza że γ w powyższym równaniu jest równe zero.
Trzecia strategia (Ide Dec hi) pozwala tylko na limitowany feedback z użyciem jednego, najbardziej podobnego dokumentu oznaczonego jako nierelewantny.

10. Ukryta analiza semantyczna

10.1. Definicja ukrytej analizy semantycznej

Ukryta analiza semantyczna (ang. latent semantic analysis) jest techniką odkrywania ukrytych zależności w zbiorze dokumentów tekstowych. Powstała ona w odpowiedzi na narastające problemy wyszukiwania dokumentów relewantnych z użyciem słów podawanych przez użytkownika. Głównym celem użytkownika przy wyszukiwaniu informacji nie jest znalezienie dokumentów posiadających podane słowa, lecz znalezienie dokumentów, które są związane z danym konceptem kryjącym się za podanymi słowami. [15]

Ukryta analiza semantyczna polega na dekompozycji macierzy term-dokument według wartości szczególnych. Pozwala to na dekompozycję oryginalnej macierzy do liniowo niezależnych obiektów, które pozwalają na dokładniejsze zbadanie zależności pomiędzy dokumentami i zredukowanie nieistotnych szumów. Przeciętna macierz term-dokument może zawierać nawet dziesiątki tysięcy kolumn i wierszy. Dekompozycja macierzy według wartości własnych pozwala na skonstruowanie aproksymowanej macierzy term-dokument dużo niższego stopnia. Następnie korzystając z aproksymowanej macierzy, można wyliczyć podobieństwo pomiędzy wektorami, zarówno dla dokumentów jak i termów.

Najważniejszym efektem ukrytej analizy semantycznej jest zwiększenie wartości zwrotu, ponieważ ukryta analiza semantyczna pozwala rozwiązać problem synonimów oraz polisemii, które są typowymi problemami w przetwarzaniu języka naturalnego z użyciem modelu przestrzeni wektorowej. Również wartość precyzji jest większa, pomimo stosowania aproksymacji macierzy niższego rzędu.

Dodatkową zaletą ukrytej analizy semantycznej jest możliwość indeksowania dokumentów jednocześnie w różnych językach (ang. cross-language information retrieval). Dzięki temu, podając zapytanie w jednym języku, otrzymamy również dokumenty relewantne z innych języków. Główną wadą ukrytej analizy semantycznej jest ogromna złożoność obliczeniowa, w porównaniu do innych technik wyszukiwania informacji. Również wyzwaniem jest dobór odpowiedniego stopnia aproksymowanej macierzy (dobór ilości rozpatrywanych wymiarów), ponieważ jego wartość wpływa na odpowiednie wyostrzenie wyszukiwania ukrytych zależności.

Macierz term-dokument A (o wymiarach $m \times n$) zostanie rozłożona na iloczyn trzech macierzy, macierz ortogonalną U (o wymiarach $m \times m$), macierz diagonalną S (o wymiarach $m \times n$) oraz transpozycję macierzy ortogonalnej V (o wymiarach $n \times n$). [22]. Dekompozycję tą można opisać wzorem:

$$A = USV^T$$

Kolumny macierzy U są ortonormalnymi wektorami własnymi macierzy AA^T, natomiast kolumny macierzy V są ortonormalnymi wektorami własnymi macierzy A^TA.

Zachodzi zależność:

$$U^T U = I$$
$$V^T V = I$$

Macierz diagonalna S zawiera w kolejności malejącej pierwiastki z wartości własnych, odpowiadającym odpowiednim wektorom własnym z kolumn z macierzy U lub V.

10.2. Przykład wyszukiwania informacji z użyciem ukrytej analizy semantycznej

Poniżej zostanie zaprezentowany przykład wyszukiwania informacji z użyciem ukrytej analizy semantycznej. Przyjmijmy, że istnieje zbiór dokumentów o następujących tytułach:

d1: Fundusze narodowe

d2: Finanse państwowe

d3: Finanse narodowe

d4: Europa w Unii

d5: Polityka Unii Europejskiej

d6: Strategia narodowa w Europie

Poniżej zdefiniowano macierz term-dokument, gdzie każda kolumna odpowiada odpowiedniemu dokumentowi, a każdy wiersz odpowiedniemu termowi. Dokumenty poddano normalizacji oraz usunięto słowa nieznaczące. Każda komórka poniższej macierzy A[i,j] oznacza ilość wystąpień termu *i* w tytule dokumentu *j*.

	*d*1	*d*2	*d*3	*d*4	*d*5	*d*6
Finanse	0	1	1	0	0	0
Fundusze	1	0	0	0	0	0
Państwo	0	1	0	0	0	0
Naród	1	0	1	0	0	1
Polityka	0	0	0	0	1	0
Strategia	0	0	0	0	0	1
Europa	0	0	0	1	1	1
Unia	0	0	0	1	1	0

Tab. 10.1: Macierz A[*i,j*] z ilością wystąpień termu *i* w tytule dokumentu *j*. Opracowanie własne.

Celem zapytania jest znalezienie dokumentów o tematyce funduszy narodowych. Zapytanie te można wyrazić jako:

q: 'fundusze' oraz 'naród'

Poniżej zaprezentowano macierz A oraz transponowaną macierzy A:

$$A=\begin{bmatrix}0&1&1&0&0&0\\1&0&0&0&0&0\\0&1&0&0&0&0\\1&0&1&0&0&1\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&1&1&1\\0&0&0&1&1&0\end{bmatrix} \quad A^T=\begin{bmatrix}0&1&0&1&0&0&0&0\\1&0&1&0&0&0&0&0\\1&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&1\\0&0&0&0&1&0&1&1\\0&0&0&1&0&1&1&0\end{bmatrix}$$

Poniżej zaprezentowano iloczyny macierzy A oraz transponowanej macierzy A. Macierze te są kwadratowe oraz symetryczne. Macierz AA^T definiuje się również innym terminem jako macierz term-term oraz odpowiednio macierz A^TA definiuje się jako macierz dokument-dokument.

$$AA^T=\begin{bmatrix}0&1&1&0&0&0\\1&0&0&0&0&0\\0&1&0&0&0&0\\1&0&1&0&0&1\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&1&1&1\\0&0&0&1&1&0\end{bmatrix}\begin{bmatrix}0&1&0&1&0&0&0&0\\1&0&1&0&0&0&0&0\\1&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&1\\0&0&0&0&1&0&1&1\\0&0&0&1&0&1&1&0\end{bmatrix}=\begin{bmatrix}2&0&1&1&0&0&0&0\\0&1&0&1&0&0&0&0\\1&0&1&0&0&0&0&0\\1&1&0&3&0&1&1&0\\0&0&0&0&1&0&1&1\\0&0&0&1&0&1&1&0\\0&0&0&1&1&1&3&2\\0&0&0&0&1&0&2&2\end{bmatrix}$$

$$A^TA=\begin{bmatrix}0&1&0&1&0&0&0&0\\1&0&1&0&0&0&0&0\\1&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&1\\0&0&0&0&1&0&1&1\\0&0&0&1&0&1&1&0\end{bmatrix}\begin{bmatrix}0&1&1&0&0&0\\1&0&0&0&0&0\\0&1&0&0&0&0\\1&0&1&0&0&1\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&1&1&1\\0&0&0&1&1&0\end{bmatrix}=\begin{bmatrix}2&0&1&0&0&1\\0&2&1&0&0&0\\1&1&2&0&0&1\\0&0&0&2&2&1\\0&0&0&2&3&1\\1&0&1&1&1&3\end{bmatrix}$$

Następnie zgodnie z poniższym wzorem, dokonano dekompozycji macierzy:

$$A=USV^T$$

Macierz A o wymiarach m × n może mieć co najwyżej min(m,n) wartości własnych, dlatego poniżej dokonano uproszczonego zapisu powyższej dekompozycji:

$$U=\begin{bmatrix}0{,}127&-0{,}44&-0{,}69&0{,}025&-0{,}299&-0{,}101\\0{,}094&-0{,}232&0{,}219&-0{,}624&0{,}535&0{,}018\\0{,}027&-0{,}154&-0{,}519&0{,}115&0{,}67&0{,}171\\0{,}435&-0{,}661&0{,}291&-0{,}135&-0{,}238&-0{,}011\\0{,}244&0{,}227&-0{,}134&-0{,}271&-0{,}235&0{,}862\\0{,}242&-0{,}143&0{,}243&0{,}579&0{,}197&0{,}243\\0{,}686&0{,}253&0{,}032&0{,}26&0{,}151&-0{,}132\\0{,}444&0{,}396&-0{,}211&-0{,}319&-0{,}046&-0{,}375\end{bmatrix}$$

$$S=\begin{bmatrix}2{,}374&0&0&0&0&0\\0&1{,}962&0&0&0&0\\0&0&1{,}526&0&0&0\\0&0&0&1{,}103&0&0\\0&0&0&0&0{,}744&0\\0&0&0&0&0&0{,}641\end{bmatrix}$$

$$V^T=\begin{bmatrix}0{,}223&0{,}065&0{,}237&0{,}476&0{,}578&0{,}574\\-0{,}455&-0{,}303&-0{,}561&0{,}33&0{,}446&-0{,}281\\0{,}334&-0{,}792&-0{,}261&-0{,}117&-0{,}205&0{,}371\\-0{,}688&0{,}127&-0{,}1&-0{,}053&-0{,}299&0{,}639\\0{,}398&0{,}499&-0{,}722&0{,}141&-0{,}175&0{,}146\\0{,}012&0{,}11&-0{,}174&-0{,}793&0{,}552&0{,}156\end{bmatrix}$$

Następnie dokonano redukcji wymiarów macierzy do k=2. Redukcja ta pozwoli na wyostrzenie zależności pomiędzy termami oraz dokumentami.

$$U_k=\begin{bmatrix}0{,}127 & -0{,}44\\ 0{,}094 & -0{,}232\\ 0{,}027 & -0{,}154\\ 0{,}435 & -0{,}661\\ 0{,}244 & 0{,}227\\ 0{,}242 & -0{,}143\\ 0{,}686 & 0{,}253\\ 0{,}444 & 0{,}396\end{bmatrix}\quad S_k=\begin{bmatrix}2{,}374 & 0\\ 0 & 1{,}962\end{bmatrix}\quad V_k^T=\begin{bmatrix}0{,}223 & 0{,}065 & 0{,}237 & 0{,}476 & 0{,}578 & 0{,}574\\ -0{,}455 & -0{,}303 & -0{,}561 & 0{,}33 & 0{,}446 & -0{,}281\end{bmatrix}$$

$$A_k=U_kS_kV_k^T$$

Poniżej zaprezentowaną aproksymowaną macierz A_k:

$$A_k=\begin{bmatrix}0{,}46 & 0{,}281 & 0{,}556 & -0{,}142 & -0{,}211 & 0{,}416\\ 0{,}256 & 0{,}152 & 0{,}308 & -0{,}044 & -0{,}074 & 0{,}256\\ 0{,}152 & 0{,}096 & 0{,}185 & -0{,}069 & -0{,}097 & 0{,}122\\ 0{,}82 & 0{,}46 & 0{,}972 & 0{,}064 & 0{,}019 & 0{,}957\\ -0{,}074 & -0{,}097 & -0{,}113 & 0{,}422 & 0{,}533 & 0{,}207\\ 0{,}256 & 0{,}122 & 0{,}293 & 0{,}18 & 0{,}207 & 0{,}408\\ 0{,}137 & -0{,}044 & 0{,}108 & 0{,}938 & 1{,}162 & 0{,}795\\ -0{,}118 & -0{,}167 & -0{,}186 & 0{,}758 & 0{,}956 & 0{,}387\end{bmatrix}$$

Po przemnożeniu odpowiednich macierzy, otrzymujemy nową reprezentacje termów oraz dokumentów będących k-wymiarowymi wektorami. Wektory te można ze sobą porównywać w celu wykrycia ukrytych zależności.

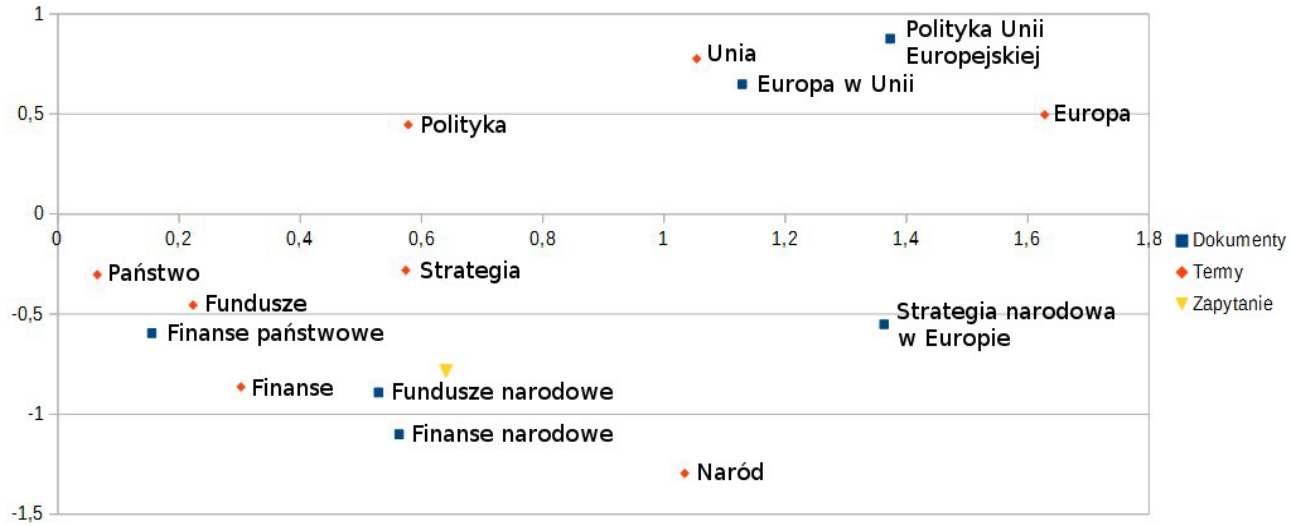
$$\begin{matrix}\textit{Finanse}\\ \textit{Fundusze}\\ \textit{Państwo}\\ \textit{Naród}\\ \textit{Polityka}\\ \textit{Strategia}\\ \textit{Europa}\\ \textit{Unia}\end{matrix}\quad U_kS_k=\begin{bmatrix}0{,}302 & -0{,}864\\ 0{,}223 & -0{,}455\\ 0{,}065 & -0{,}303\\ 1{,}034 & -1{,}296\\ 0{,}578 & 0{,}446\\ 0{,}574 & -0{,}281\\ 1{,}628 & 0{,}496\\ 1{,}054 & 0{,}776\end{bmatrix}$$

$$\begin{matrix}\textit{Fundusze narodowe}\\ \textit{Finanse państwowe}\\ \textit{Finanse narodowe}\\ \textit{Europa w Unii}\\ \textit{Polityka Unii Europejskiej}\\ \textit{Strategia narodowa w Europie}\end{matrix}\quad (S_kV_k^T)^T=\begin{bmatrix}0{,}529 & -0{,}892\\ 0{,}155 & -0{,}595\\ 0{,}563 & -1{,}101\\ 1{,}129 & 0{,}648\\ 1{,}373 & 0{,}876\\ 1{,}363 & -0{,}551\end{bmatrix}$$

Zapytanie 'fundusze' oraz 'naród' może zostać utworzone poprzez utworzenie centroidu z użyciem algorytmu Rocchio:

$$q=\frac{1}{2}(\begin{bmatrix}0{,}223 & -0{,}455\end{bmatrix}+\begin{bmatrix}1{,}034 & -1{,}296\end{bmatrix})=\begin{bmatrix}0{,}629 & -0{,}876\end{bmatrix}$$

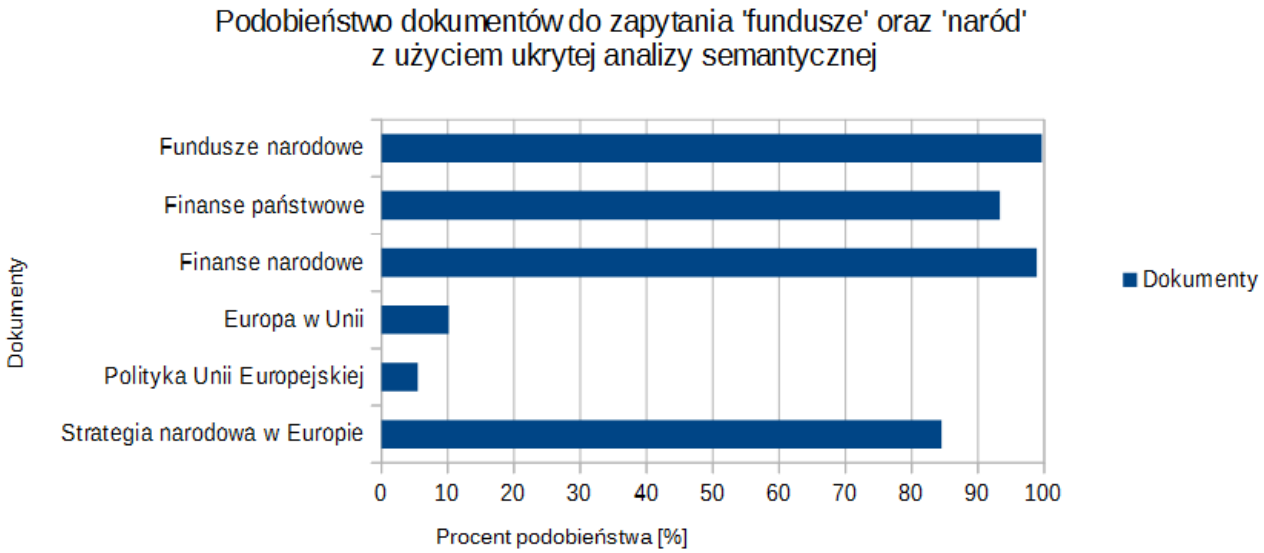
Na rys. 10.1 zaprezentowano geometryczną reprezentację dokumentów, termów oraz zapytania *q*.



Rys. 10.1: Geometryczna reprezentacja dokumentów wraz z ich termami oraz zapytania 'fundusze' oraz 'naród'

Na rys. 10.1 można zaobserwować, że termy i dokumenty o tematyce finansowej są blisko siebie, tak samo jak i termy i dokumenty o tematyce Unii Europejskiej. Warto również zauważyć, że term 'Polityka' jest bardzo blisko termu 'Strategia'.

Podobieństwo dokumentów do zapytania *q* może zostać wyrażone przy pomocy kosinusoidalnej miary podobieństwa. Na rys. 10.2 zaprezentowano podobieństwo każdego z dokumentów do zapytania 'fundusze' oraz 'naród'.



Rys. 10.2: Podobieństwo dokumentów do zapytania 'fundusze' oraz 'naród' z użyciem ukrytej analizy semantycznej. Opracowanie własne.

Dokument 'Finanse państwowe' posiada 93,3% podobieństwa do wskazanego zapytania, pomimo tego, że jego tytuł nie zawiera żadnego z termów wymienionych w zapytaniu.

11. Przeprowadzone badania

11.1. Cel badań

Celem badań było porównanie różnych strategii wyszukiwania informacji z użyciem modelu przestrzeni wektorowej. Zostały porównane następujące strategie:

- Wyszukiwanie standardowe przy pomocy kosinusoidalnej miary podobieństwa
  - Porównanie technik normalizacji dokumentów tekstowych: słownik morfologiczny Ispell oraz algorytm Snowball stemming
  - Wpływ techniki Tf-idf na otrzymywane wyniki
- Wyszukiwanie przy pomocy najczęściej występujących termów
  - Rozkład ilości występowania identycznych termów dla każdej pary dokumentów.
  - Wyniki wyszukiwania w zależności od ilości uwzględnionych wspólnych termów
- Optymalizacja wyszukiwania z wykorzystaniem sprzężenia zwrotnego
  - Znalezienie optymalnych współczynników dla sprzężenia zwrotnego
  - Wpływ sprzężenia zwrotnego na otrzymywane wyniki
- Wyszukiwanie w oparciu o centroidy użytkowe – klasyfikacja binarna oraz wieloklasowa
  - Rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego
  - Wyniki wyszukiwania w oparciu o centroidy użytkowe
- Optymalizacja wyszukiwania z użyciem ukrytej analizy semantycznej
  - Wpływ wielkości macierzy term-dokument na czas jej dekompozycji
  - Wyniki wyszukiwania z wykorzystaniem ukrytej analizy semantycznej
  - Rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego z wykorzystaniem ukrytej analizy semantycznej
  - Wyszukiwanie w oparciu o centroidy użytkowe z wykorzystaniem ukrytej analizy semantycznej – klasyfikacja binarna oraz wieloklasowa

Przeprowadzone badania zostały zaimplementowane przy użycia języka Java w wersji 8, w środowisku programistycznym Eclipse Kepler SR 2. Najnowsza wersja języka Javy w wersji 8, pozwoliła na skorzystanie ze strumieni (java.util.stream) oraz wyrażeń lambda, które znacząco poprawiły łatwość implementacji powyższych strategii jak i umożliwiły prostą paralelizację obliczeń z użyciem równoległych strumieni (ang. parallel streams). Do przechowywania danych użyto popularnej open sourcowej bazy danych PostgreSQL 9.3. Baza ta w sposób zaawansowany wspiera proces tokenizacji oraz normalizacji dokumentów tekstowych. Wbudowany jest algorytm stemmingu Snowball oraz możliwe jest stosowanie różnego rodzaju słowników – stop list, słowników morfologicznych oraz tezaurusów. Możliwe jest również wykonywanie operacji na typie tsvector, reprezentującym dokument w postaci wektora termów. [17]

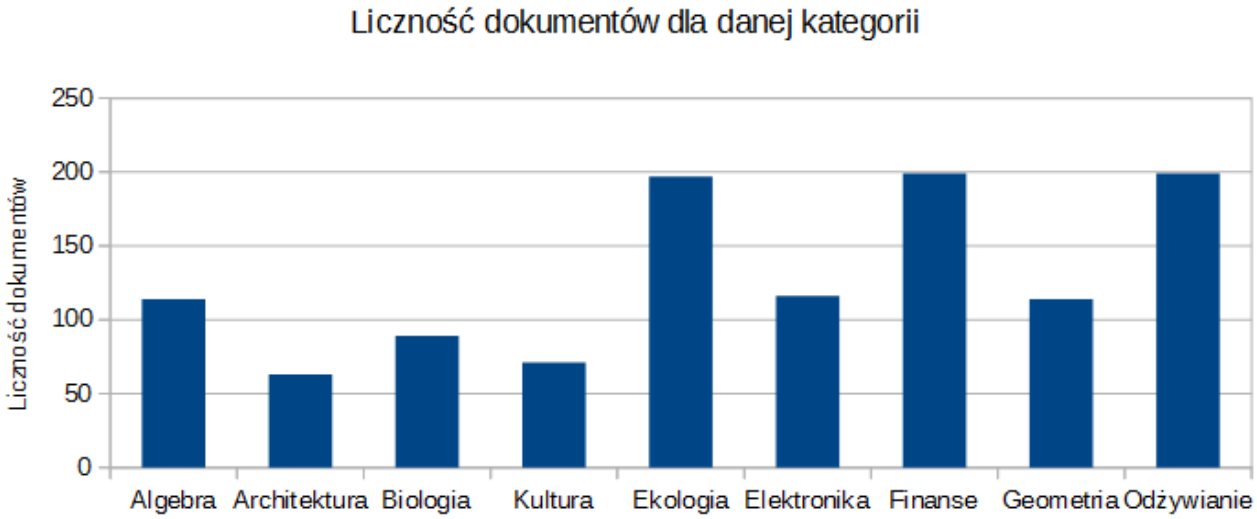
W ramach jednej konfiguracji algorytmu oraz ustalonych danych wejściowych, algorytm był deterministyczny i zwracał zawsze ten sam wynik, dlatego nie było potrzeby wykonywania serii testów dla tej samej konfiguracji. Zamiast tego badania zawsze wykonywano w izolowanym środowisku dla każdego dokumentu (gdzie zapytaniem był dany dokument), a wyniki były uśredniane dla całej kolekcji.

Funkcje oceny wyszukiwania informacji, precyzja oraz zwrot, konkurują wzajemnie ze sobą, dlatego do końcowego porównania różnych strategii, została użyta F-miara, która jest średnią harmoniczną obu powyższych funkcji. Każda strategia posiadała odpowiednie parametry konfiguracyjne, co wpłynęło na wyniki uzyskiwane przez daną strategię. Dlatego w końcowym badaniu zostały porównane ze sobą wszystkie strategie z uzyskaną najlepszą możliwą dla niej konfiguracją, pozwalającą na uzyskanie najwyższej możliwej wartości średniej F-miary. Badania porównujące strategie wyszukiwania i rekomendacji informacji zostały przeprowadzone zgodnie z metodyką Cranfield.

11.2. Dane testowe

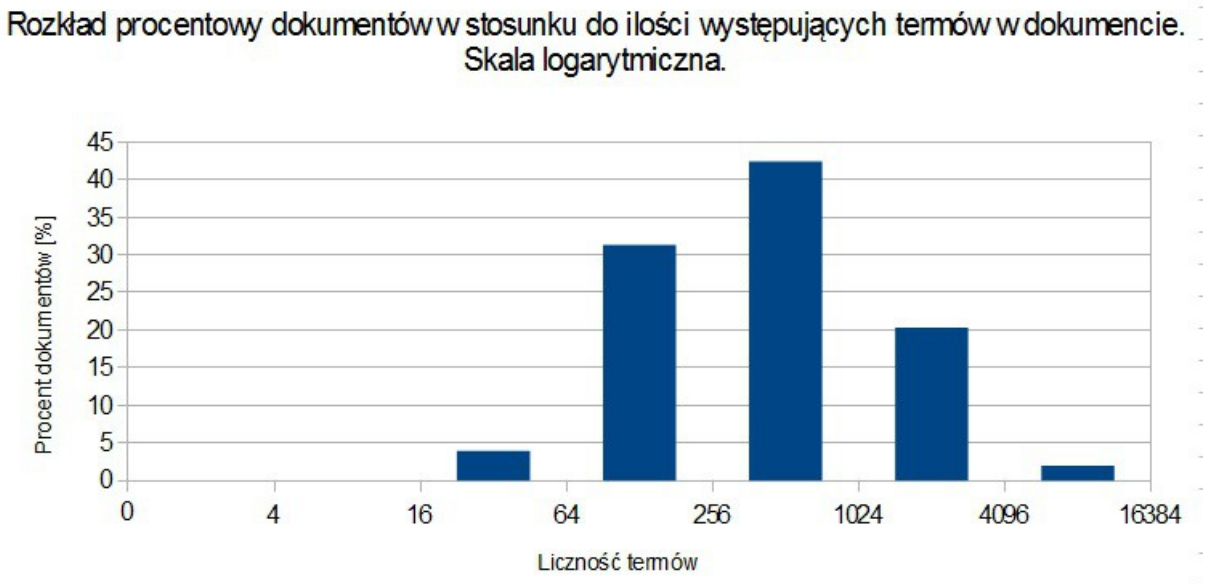
Do badań użyto artykułów z Wikipedii anglojęzycznej, które zostały pobrane i wyekstrahowane z użyciem biblioteki jsoup [23]. Z portalu tematycznego [24] wybrano 9 kategorii, gdzie dla każdej kategorii pobrano najważniejsze artykuły opisujące daną tematykę, łącznie 1162 artykułów. Do badań użyto następujących kategorii:
Algebra, Żywienie, Biologia, Ekologia, Geometria, Architektura, Kultura, Finanse, Elektronika.

Artykuły te symulowały osobiste dane tekstowe zgromadzone przez użytkownika w systemie do zarządzania informacją osobistą. Symulacja ta odpowiadała zgromadzeniu przez użytkownika różnych artykułów z interesujących go zagadnień tematycznych. Informacje te były porównywane z użyciem modelu przestrzeni wektorowej. Na rys. 11.1 przedstawiono liczność dokumentów dla każdej kategorii:



Rys. 11.1: Liczność dokumentów dla danej kategorii. Opracowanie własne.

Aby lepiej zobrazować specyfikę dokumentów występujących w badanej kolekcji, na rys. 11.2 przedstawiono rozkład procentowy dokumentów do ilości występujących termów w dokumencie. Ponad 40% dokumentów w całej kolekcji posiadało liczbę wyekstrahowanych termów w przedziale [256,1024]. Przeciętny dokument posiadał średnio 788 termów.

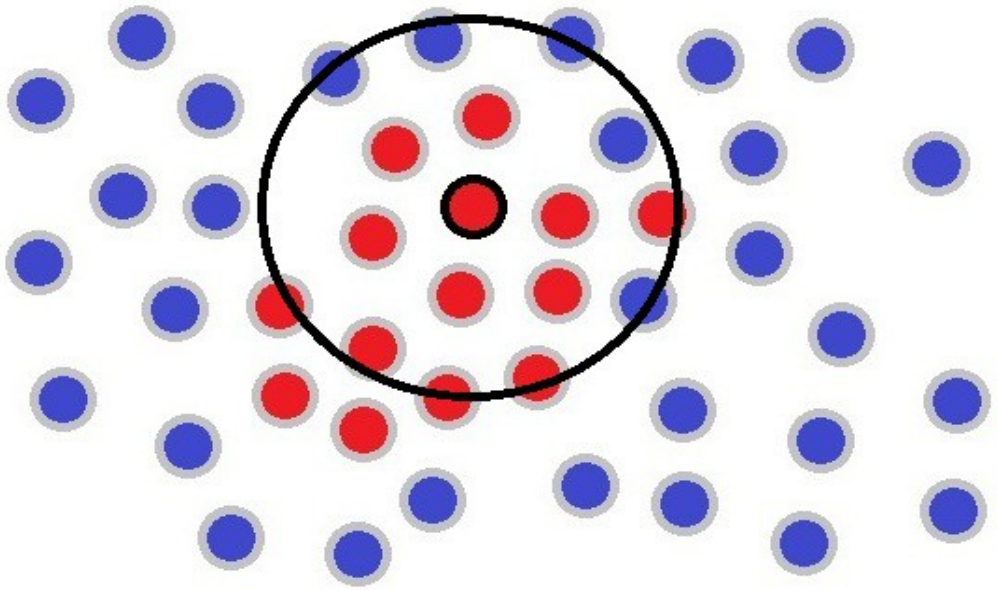


Rys. 11.2: Rozkład procentowy dokumentów w stosunku do ilości występujących termów w dokumencie. Opracowanie własne.

11.3. Opis założeń

Przyjęto założenie, że użytkownik posiadał kolekcję dokumentów tekstowych oraz nie posiadał żadnej kategoryzacji swoich dokumentów (nie dokonał żadnej klasyfikacji swoich dokumentów). Użytkownik przeglądając dowolny jeden dokument, chciałby znaleźć również inne dokumenty o podobnej tematyce. Należało przyjąć założenie, że dowolna para dokumentów z tej samej kategorii dokumentów była zawsze w pewnym stopniu dla siebie relewantna.

Użytkownik przeglądając dowolny dokument, nie wiedział jak w przestrzeni wektorowej dany dokument był podobny do innych dokumentów z danej tematyki. Nieznana była odległość dokumentu wybranego przez użytkownika od centroidu danej tematyki. Model przestrzeni wektorowej umożliwił wyliczenie podobieństwa dla dowolnych dokumentów, zatem istniała możliwość utworzenia rankingu dokumentów najbardziej podobnych do dokumentu wskazanego przez użytkownika. Dokumenty powyżej ustalonej wartości podobieństwa (wyrażonej w procentach) zostały zaproponowane użytkownikowi jako relewantne. Celem systemu było zwrócenie jak największej ilości dokumentów relewantnych (rekomendacji prawdziwie pozytywnych), z możliwie jak najmniejszą ilością dokumentów nierelewantnych (rekomendacji fałszywie pozytywnych). Na rys. 11.3 zaprezentowano ideę działania powyższego algorytmu.



Rys. 11.3: Zadanie maksymalizacji rekomendacji prawdziwie pozytywnych z możliwie jak najmniejszą ilością rekomendacji fałszywie pozytywnych. Rysunek poglądowy dla wektorów w przestrzeni R^2-wymiarowej. Opracowanie własne.

11.4. Wyszukiwanie standardowe przy pomocy kosinusoidalnej miary podobieństwa

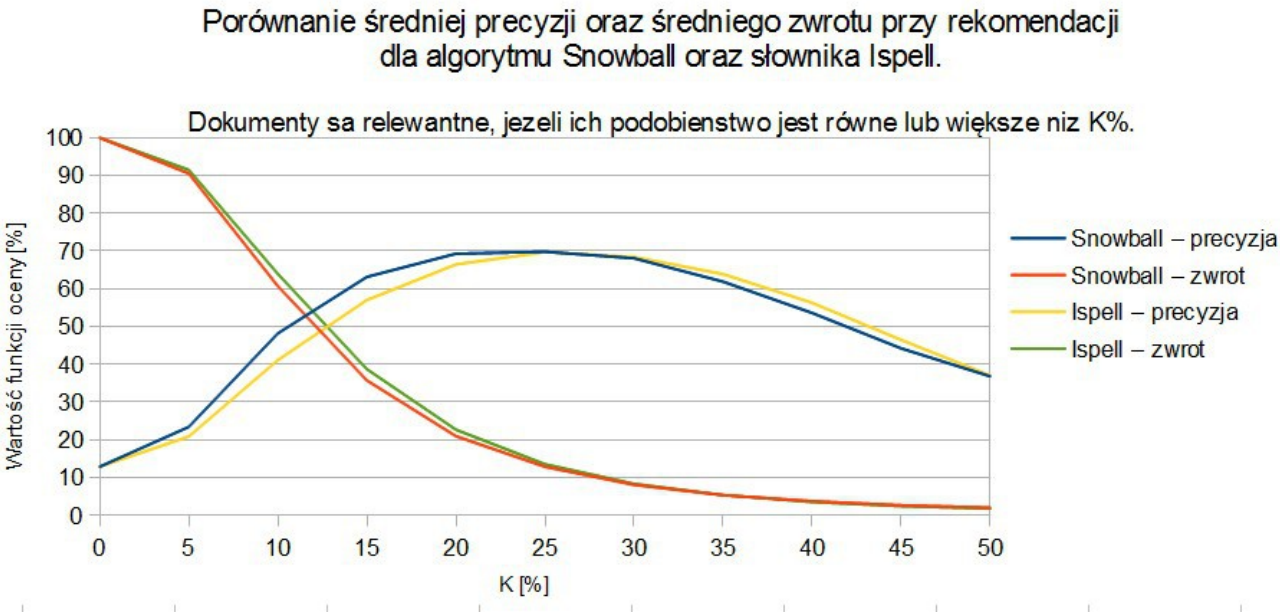
W ramach jednej iteracji algorytmu, badanie zostało przeprowadzone dla wszystkich dokumentów, w wyniku czego powstały uśrednione wartości precyzji i zwrotu. Algorytm wykonał po jednej iteracji dla każdej możliwej wartości parametru K w przedziale [0,50]. Dodatkowo, każda iteracja została przeprowadzona osobno z wykorzystaniem algorytmu stemmingu Snowball oraz osobno z wykorzystaniem techniki lematyzacji z słownikiem morfologicznym Ispell.

Schemat badania został opisany w postaci pseudokodu, opisanego na rys. 11.4

```
Dla każdego k:
    Dla każdego dokumentu d w kolekcji:
        Zwróć wszystkie dokumenty różne od d, jeżeli podobieństwo >= k
        Wyznacz precyzję oraz zwrot dla kategorii dokumentu d
    Policz średnią precyzję oraz średni zwrot dla wszystkich dokumentów
```

Rys. 11.4: Pseudokod dla badania porównania średniej precyzji oraz średniego zwrotu przy rekomendacji dla algorytmu Snowball oraz słownika Ispell w zależności od K, gdy dokumenty są przyjęte za relewantne, jeżeli ich podobieństwo większe niż K%. Opracowanie własne.

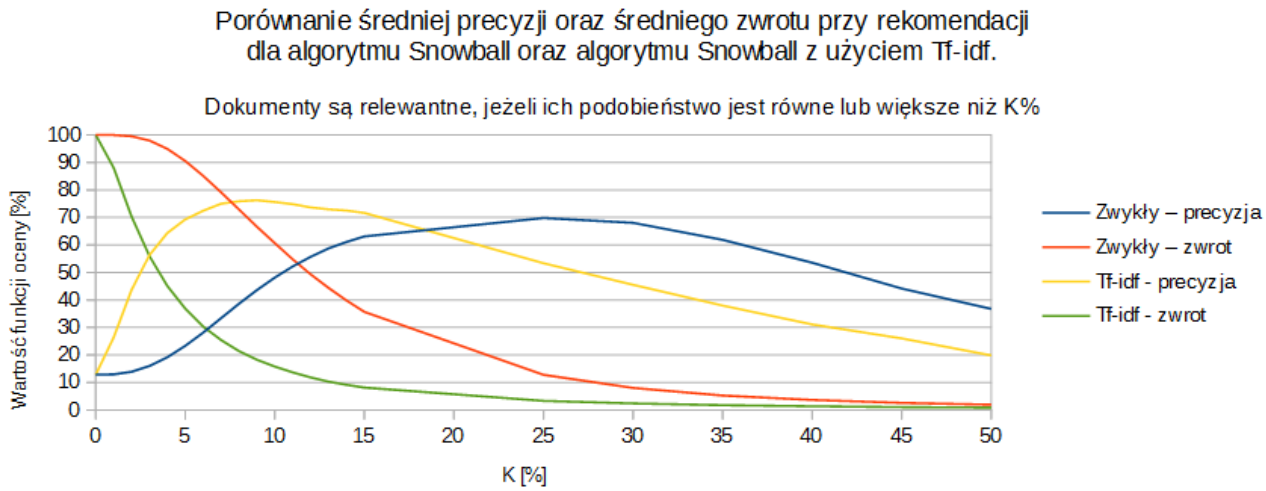
Wyniki badania zostały przedstawione na rys. 11.5



Rys. 11.5: Porównanie średniej precyzji oraz średniego zwrotu przy rekomendacji dla algorytmu Snowball oraz słownika Ispell. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K%. Opracowanie własne.

Z wykonanych badań (rys. 11.5) wynika, że zarówno technika stemmingu jak i lematyzacji, osiągnęła podobne wyniki. Dla K równego 20, algorytm stemmingu Snowball uzyskał wartość 69,27% średniej precyzji przy 20,91% średnim zwrocie, z kolei algorytm lematyzacji z słownikiem Ispell dla tej samej wartości K uzyskał 66,43% średniej precyzji dla wartości 22,63% średniego zwrotu. Warto zauważyć, że dla K powyżej 25% dla obu algorytmów wartości średniej precyzji zaczynały maleć, choć zdawało by się, że przy tak silnym kryterium podobieństwa średnia precyzja była by bardzo wysoka. Było to związane z faktem, że dokumenty nawet o wysokim podobieństwie do wskazanego dokumentu (powyżej 25%) mogły być wciąż rzeczywiście nierelewantne dla danego dokumentu. Oczywiście dla K równego 0%, wartości zwrotu dla obu algorytmów były równe 100%, ponieważ w tym przypadku system kwalifikował wszystkie dokumenty jako relewantne.

Dalsze rozważania były kontynuowane tylko z użyciem algorytmu stemmingu Snowball.
W następnym kroku dokonano badania z użyciem techniki Tf-idf. Przed wyznaczeniem kosinusoidalnej miary podobieństwa przy każdym porównywaniu wektorów dwóch różnych dokumentów, wartości termów były przemnażane przez odpowiedni współczynnik *idf* dla danego termu, uwzględniający przydatność danego termu do rozróżniania treści różnych dokumentów. Schemat badania był identyczny z poprzednim. Badanie przeprowadzono dla porównania wyników przy zastosowaniu techniki Tf-idf lub jej braku (algorytmu zwykłego). W obu testach użyto algorytmu stemmingu Snowball. Wyniki badania zostały przedstawione na rys. 11.6.



Rys. 11.6: Porównanie średniej precyzji oraz średniego zwrotu przy rekomendacji dla algorytmu Snowball oraz algorytmu Snowball z użyciem Tf-idf. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K%. Opracowanie własne.

Zaletą techniki Tf-idf jest skupienie się na termach rzadko występujących w kolekcji, a zatem posiadających wysoką wartość dyskryminacyjną. Z wykonanych badań wynika, że technika Tf-idf rzeczywiście pozwoliła na dokładniejsze rozróżnianie dokumentów, a zatem także na wzrost średniej precyzji. Przykładowo, dla K równego 5%, technika Tf-idf osiągnęła średnią precyzję równą 69,30% przy 36,84% zwrocie. Dla porównania, algorytm nie wykorzystujący techniki Tf-idf dla K równego 20% osiągnął porównywalną średnią precyzję 69,27%, lecz przy zwrocie 20,91%.

Najwyższa możliwa wartość średniej F-miary dla powyższej rekomendacji wyniosła 53,69% oraz 56,11% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.5. Wyszukiwanie przy pomocy najczęściej występujących termów

Dokumenty tekstowe poddano również analizie pod względem najczęściej występujących termów. Dla każdej pary dokumentów wyznaczono dwa zbiory z 5 termami o największych wagach. Schemat badania został opisany w postaci pseudokodu, opisanego na rys. 11.7

```
Dla każdej pary dokumentów d1 i d2
        Wyznacz zbiór 5 termów o największych wagach z dokumentu d1
        Wyznacz zbiór 5 termów o największych wagach z dokumentu d2
        Wylicz ilość wspólnych termów występujących w obu zbiorach
Pogrupuj wyniki według ilości wspólnych termów
```

Rys. 11.7: Pseudokod do utworzenia rozkładu ilości występowania identycznych termów dla każdej pary dokumentów. 5 najczęściej występujących termów jest porównywane dla każdej pary dokumentów. Opracowanie własne.

Badanie przeprowadzono osobno z wykorzystaniem techniki Tf-idf lub jej braku. Wyniki powyższego badania zostały zaprezentowane na rys. 11.8.

Rys. 11.8: Rozkład ilości występowania identycznych termów dla każdej pary dokumentów. 5 najczęściej występujących termów jest porównywane dla każdej pary dokumentów. Opracowanie własne.

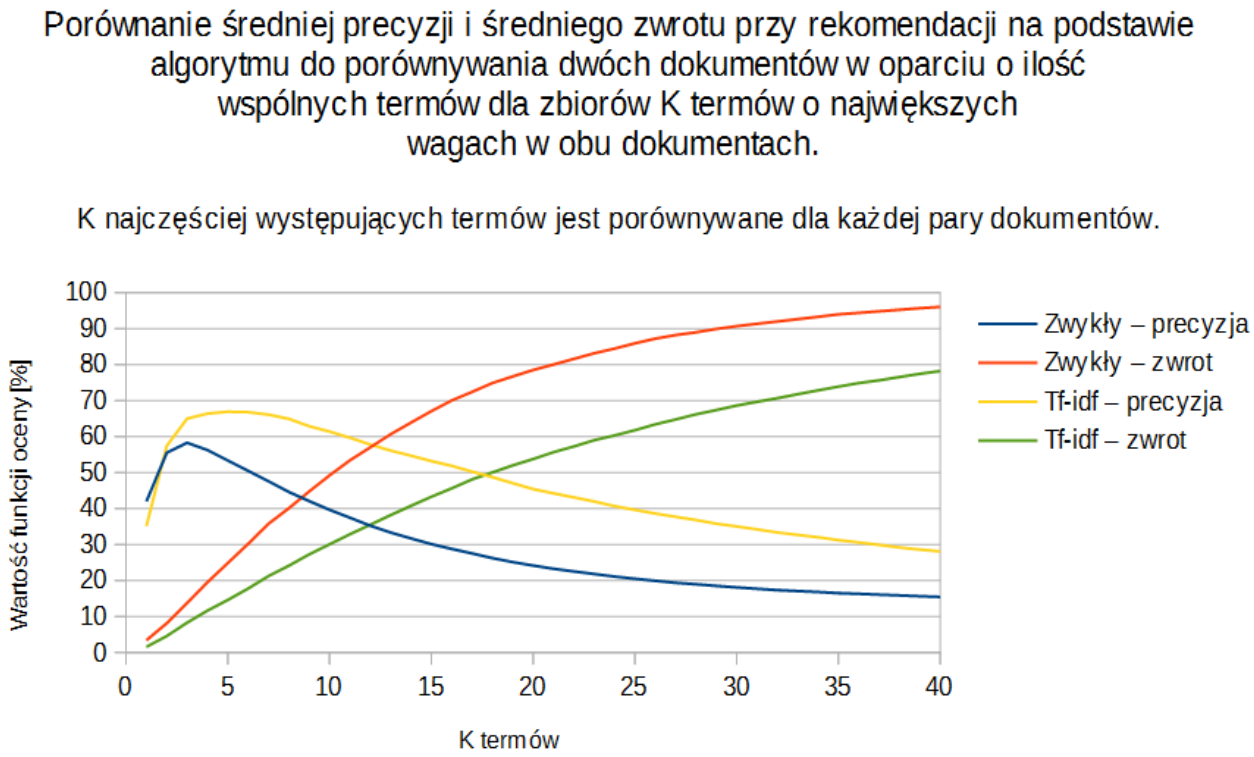
Z powyższego badania wynika, że dla obu algorytmów aż ponad 93% par dokumentów nie posiadało żadnego wspólnego termu – uwzględniając 5 termów o największych wagach w każdej parze dokumentów. Na podstawie tej obserwacji, można było wysunąć hipotezę, że dokumenty posiadające jeden lub więcej wspólnych termów mogły być relewantne dla siebie.

Z powodu powyższej hipotezy, zaproponowano algorytm do wyznaczania dokumentów relewantnych działający w oparciu o ilość wspólnych termów dla zbiorów K termów o największych wagach w obu dokumentach. Schemat działania zaproponowanego algorytmu porównującego dwa dokumenty tekstowe został przedstawiony na rys. 11.9.

```
Q = zbiór K termów o największych wagach z dokumentu będącego zapytaniem
D = zbiór K termów o największych wagach z dokumentu porównywanego
w = ilość wspólnych termów występujących w zbiorze Q i D
Jeżeli w>=1:
        Dokumenty są relewantne
W przeciwnym przypadku:
        Dokumenty nie są relewantne
```

Rys. 11.9: Pseudokod dla algorytmu do porównywania dwóch dokumentów w oparciu o ilość wspólnych termów dla zbiorów K termów o największych wagach w obu dokumentach. Opracowanie własne.

Na rys. 11.10 zaprezentowano wyniki badania dla powyższego algorytmu.



Rys. 11.10: Porównanie średniej precyzji i średniego zwrotu przy rekomendacji na podstawie algorytmu do porównywania dwóch dokumentów w oparciu o ilość wspólnych termów dla zbiorów K termów o największych wagach w obu dokumentach. Opracowanie własne.

Funkcje precyzji osiągnęły swoje ekstremum (maksimum globalne) dla K równego 11 i 19, odpowiednio dla algorytmu zwykłego oraz dla algorytmu Tf-idf. Były to wartości graniczne, powyżej których wartości średniej precyzji zaczynały maleć dla wskazanych algorytmów, ponieważ kryterium odpowiadające za określenie podobieństwa lub jego braku pomiędzy dwoma dokumentami zaczynało być coraz mniej restrykcyjne.

Najwyższa możliwa wartość średniej F-miary dla powyższej rekomendacji wyniosła 44,03% oraz 49,40% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.6. Optymalizacja wyszukiwania z wykorzystaniem sprzężenia zwrotnego

Następnym krokiem badań było wykonanie serii testów dla sprzężenia zwrotnego przy użyciu wzoru (zdefiniowanego w rozdziale 9):

$$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$

Celem badania było wyznaczenie optymalnych współczynników α, β, γ, D_r oraz D_{nr}, które pozwoliły na najlepszą poprawę jakości wyszukiwania wyników. Do porównywania wyników, została użyta funkcja R-precyzji. Wartości dla tej funkcji zostały wyliczone wpierw dla pierwotnego zapytania i następnie porównane z wartościami dla zmodyfikowanego zapytania po wykonaniu sprzężenia zwrotnego. Przy ocenie poprawy jakości wyników, zostały pominięte te dokumenty, które zostały już wybrane przez użytkownika.

Końcowym rezultatem badań było porównanie średniego procentowego wzrostu R-precyzji, gdzie w ramach jednej iteracji zostało wykonano badanie dla wszystkich dokumentów. W ramach eksperymentów przeprowadzono 1536 iteracji, gdzie w każdej iteracji została użyta inna kombinacja współczynników. Wyniki badań zostały podzielone również pod względem stosowania techniki Tf-idf oraz jej braku.

Zbadany został wpływ wag dla dokumentów relewantnych oraz nierelewantnych oraz wpływ uwzględnienia pierwotnego zapytania lub jego braku. Dodatkowo zbadano wpływ liczności zbioru dokumentów relewantnych (przynajmniej jeden dokument) oraz liczności zbioru dokumentów nierelewantnych. Założono, że użytkownik udzielił feedbacku na temat rzeczywistej relewancji maksymalnie do 10 pierwszych dokumentów zwracanych przez system. W badaniach zbadano również wpływ stosowania normalizacji ilości dokumentów relewantnych i nierelewantnych, którą Ide odrzuciła w swoich badaniach [21]. W tab. 11.1 zaprezentowano zbiór badanych wartości dla każdego ze współczynników.

Współczynik	**Opis współczynnika**	**Zbiór**
α	Waga pierwotnego zapytania	{0; 1}
β	Waga dla dokumentów relewantnych	{0,5; 1,0; 2,0; 4,0}
γ	Waga dla dokumentów nierelewantnych	{0,5; 1,0; 2,0; 4,0}
$\|D_r\|$	Liczność zbioru dokumentów relewantnych	{1; 3; 5}
$\|D_{nr}\|$	Liczność zbioru dokumentów nierelewantnych	{0; 1; 3; 5}
Normalizacja	Normalizacja dokumentów relewantnych i nierelewantnych	{Tak; Nie}

Tab. 11.1: Zbiór wartości badanych dla każdego ze współczynników. Opracowanie własne.

W tab. 11.2 oraz tab. 11.3 zaprezentowano 25 najlepszych oraz 25 najgorszych kombinacji powyższych współczynników, posortowanych malejąco. Badania również przeprowadzono osobno z wykorzystaniem techniki Tf-idf, a wyniki zaprezentowano w tab. 11.4 i tab. 11.5.

α	β	γ	D_r	D_{nr}	Normalizacja	Wzrost R-Precyzji
1	2	4	5	5	Nie	21.32%
1	1	2	5	5	Nie	21.29%
0	2	4	5	5	Tak	21.17%
0	1	2	5	5	Tak	21.17%
0	0.5	1	5	5	Tak	21.17%
0	2	4	5	5	Nie	21.17%
0	1	2	5	5	Nie	21.17%
0	0.5	1	5	5	Nie	21.17%
1	0.5	1	5	5	Nie	20.92%
1	2	4	5	5	Tak	20.64%
1	0.5	2	5	5	Nie	20.55%
1	1	4	5	5	Nie	20.30%
1	1	4	5	5	Tak	19.89%
0	1	4	5	5	Tak	19.52%
0	0.5	2	5	5	Tak	19.52%
0	1	4	5	5	Nie	19.52%
0	0.5	2	5	5	Nie	19.52%
0	1	4	3	5	Tak	17.61%
0	0.5	2	3	5	Tak	17.61%
0	1	4	5	3	Tak	17.24%
0	0.5	2	5	3	Tak	17.24%
1	0.5	2	3	5	Nie	17.06%
1	1	4	3	5	Nie	16.51%
0	0.5	4	5	3	Nie	16.15%
1	0.5	4	5	5	Nie	16.10%

Tab. 11.2: Lista kombinacji współczynników dla sprzężenia zwrotnego uzyskujących najlepszy średni procentowy wzrost R-Precyzji. Opracowanie własne.

α	β	γ	D_r	D_{nr}	Normalizacja	Wzrost R-Precyzji
1	0.5	0.5	1	5	Tak	9.51%
1	0.5	0.5	1	3	Tak	8.97%
1	0.5	0.5	3	0	Tak	6.74%
1	0.5	1	3	0	Tak	6.74%
1	0.5	2	3	0	Tak	6.74%
1	0.5	4	3	0	Tak	6.74%
1	0.5	0.5	5	0	Tak	6.62%
1	0.5	1	5	0	Tak	6.62%
1	0.5	2	5	0	Tak	6.62%
1	0.5	4	5	0	Tak	6.62%
1	1	0.5	1	0	Nie	5.60%
1	1	1	1	0	Nie	5.60%
1	1	2	1	0	Nie	5.60%
1	1	4	1	0	Nie	5.60%
1	1	0.5	1	0	Tak	5.60%
1	1	1	1	0	Tak	5.60%
1	1	2	1	0	Tak	5.60%
1	1	4	1	0	Tak	5.60%
1	2	0.5	1	0	Nie	5.59%
1	2	1	1	0	Nie	5.59%
1	2	2	1	0	Nie	5.59%
1	2	4	1	0	Nie	5.59%
1	0.5	0.5	1	0	Nie	5.05%
1	0.5	1	1	0	Nie	5.05%
1	0.5	2	1	0	Nie	5.05%

Tab. 11.3: Lista kombinacji współczynników dla sprzężenia zwrotnego uzyskujących najgorszy średni procentowy wzrost R-Precyzji. Opracowanie własne.

α	β	γ	D_r	D_{nr}	Normalizacja	Wzrost R-Precyzji
1	2	4	5	5	Nie	16,04%
1	1	2	5	5	Nie	16,00%
0	2	4	5	5	Tak	15,93%
0	1	2	5	5	Tak	15,93%
0	0,5	1	5	5	Tak	15,93%
0	2	4	5	5	Nie	15,93%
0	1	2	5	5	Nie	15,93%
0	0,5	1	5	5	Nie	15,93%
1	0,5	1	5	5	Nie	15,73%
1	2	4	5	5	Tak	15,54%
1	0,5	2	5	5	Nie	15,20%
1	1	4	5	5	Nie	15,06%
1	1	4	5	5	Tak	14,50%
0	1	4	5	5	Nie	14,45%
0	0,5	2	5	5	Nie	14,45%
0	1	4	5	5	Tak	14,44%
0	0,5	2	5	5	Tak	14,44%
0	1	4	3	5	Tak	12,45%
0	0,5	2	3	5	Tak	12,45%
0	1	4	5	3	Tak	11,85%
0	0,5	2	5	3	Tak	11,85%
1	0,5	2	3	5	Nie	11,51%
1	0,5	4	5	5	Nie	11,02%
0	0,5	4	5	3	Nie	10,97%
1	1	4	3	5	Nie	10,88%

Tab. 11.4: Lista kombinacji współczynników dla sprzężenia zwrotnego z wykorzystaniem techniki Tf-idf uzyskujących najlepszy średni procentowy wzrost R-Precyzji. Opracowanie własne.

α	β	γ	D_r	D_{nr}	Normalizacja	Wzrost R-Precyzji
1	0.5	0.5	3	0	Tak	8.07%
1	0.5	1	3	0	Tak	8.07%
1	0.5	2	3	0	Tak	8.07%
1	0.5	4	3	0	Tak	8.07%
1	0.5	0.5	5	0	Tak	7.95%
1	0.5	1	5	0	Tak	7.95%
1	0.5	2	5	0	Tak	7.95%
1	0.5	4	5	0	Tak	7.95%
1	0.5	0.5	1	5	Tak	7.86%
1	0.5	0.5	1	3	Tak	7.47%
1	2	0.5	1	0	Nie	6.93%
1	2	1	1	0	Nie	6.93%
1	2	2	1	0	Nie	6.93%
1	2	4	1	0	Nie	6.93%
1	1	0.5	1	0	Nie	6.64%
1	1	1	1	0	Nie	6.64%
1	1	2	1	0	Nie	6.64%
1	1	4	1	0	Nie	6.64%
1	1	0.5	1	0	Tak	6.64%
1	1	1	1	0	Tak	6.64%
1	1	2	1	0	Tak	6.64%
1	1	4	1	0	Tak	6.64%
1	0.5	0.5	1	0	Nie	6.00%
1	0.5	1	1	0	Nie	6.00%
1	0.5	2	1	0	Nie	6.00%

Tab. 11.5: Lista kombinacji współczynników dla sprzężenia zwrotnego z wykorzystaniem techniki Tf-idf uzyskujących najgorszy średni procentowy wzrost R-precyzji. Opracowanie własne.

Wyniki zaprezentowane w tab. 11.2, 11.3, 11.4, 11.5 jednoznacznie wskazują, że ilość dokumentów ocenionych przez użytkownika w procesie sprzężenia zwrotnego miała wpływ na średni procentowy wzrost R-precyzji. Oznaczenie przez użytkownika małej ilości dokumentów jako relewantne oraz w szczególności brak wskazania żadnego z dokumentów jako nierelewantnego znacząco wpłynęło na średni procentowy wzrost R-precyzji. Zarówno dla algorytmu zwykłego oraz dla algorytmu Tf-idf, najlepszy średni procentowy wzrost R-precyzji został odnotowany dla liczności zbioru D_n oraz D_{nr} równej 5. Ponadto, w tab. 11.2 oraz 11.4 można zaobserwować, że dla 25 najlepszych kombinacji współczynników dla sprzężenia zwrotnego, współczynnik γ był zawsze większy od współczynnika β. Oznacza to, że dla danego zbioru testującego, dużo większe znaczenie miało przeciwdziałanie rekomendacjom fałszywie pozytywnych niż skupianie się na wzroście rekomendacji prawdziwie pozytywnych. Wyniki również pokazują, że normalizacja nie wpłynęła znacząco na średni procentowy wzrost R-precyzji. Warto również zauważyć w tab. 11.3 oraz 11.5, że 25 najgorszych wyników zawsze uwzględniało pierwotne zapytanie (współczynnik α był równy 1)

Zarówno dla algorytmu zwykłego oraz algorytmu Tf-idf, najlepszy średni procentowy wzrost R-precyzji został odnotowany dla następujących współczynników (tab. 11.6):

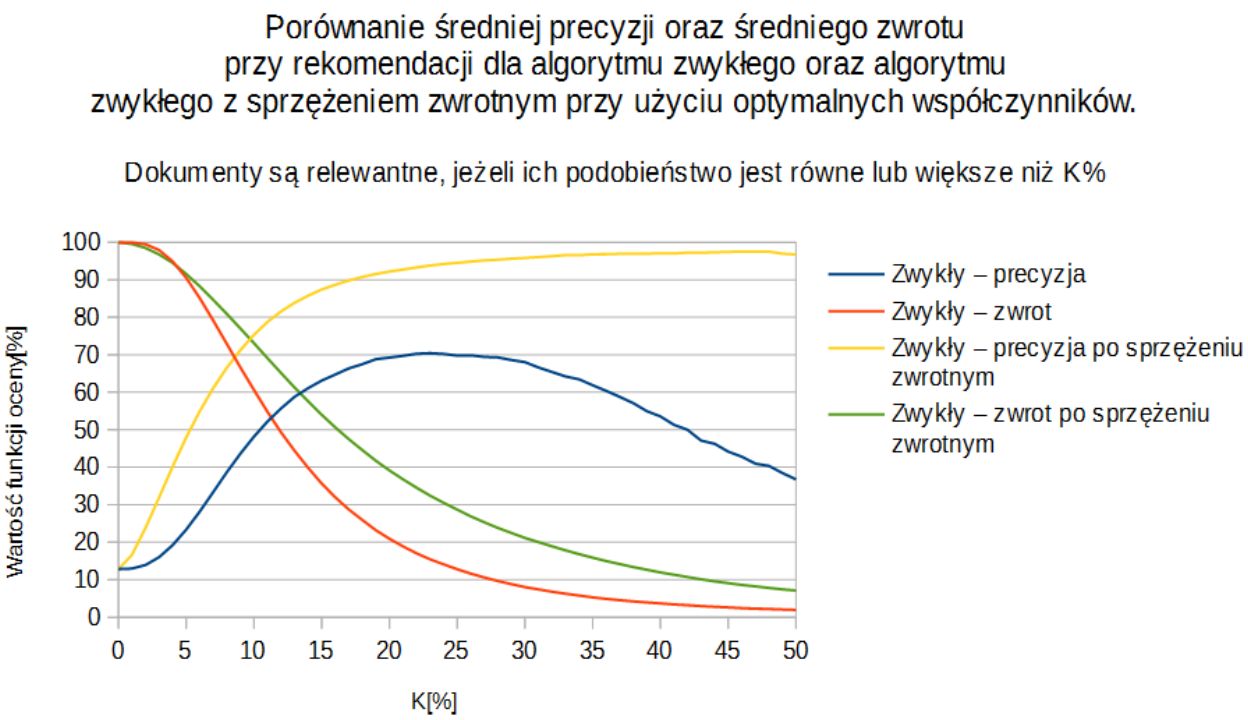
α	1
β	2
γ	4
D_r	5
D_{nr}	5
Normalizacja	Nie

Tab. 11.6: Kombinacja współczynników dla sprzężenia zwrotnego uzyskująca najlepszy średni procentowy wzrost R-precyzji. Opracowanie własne.

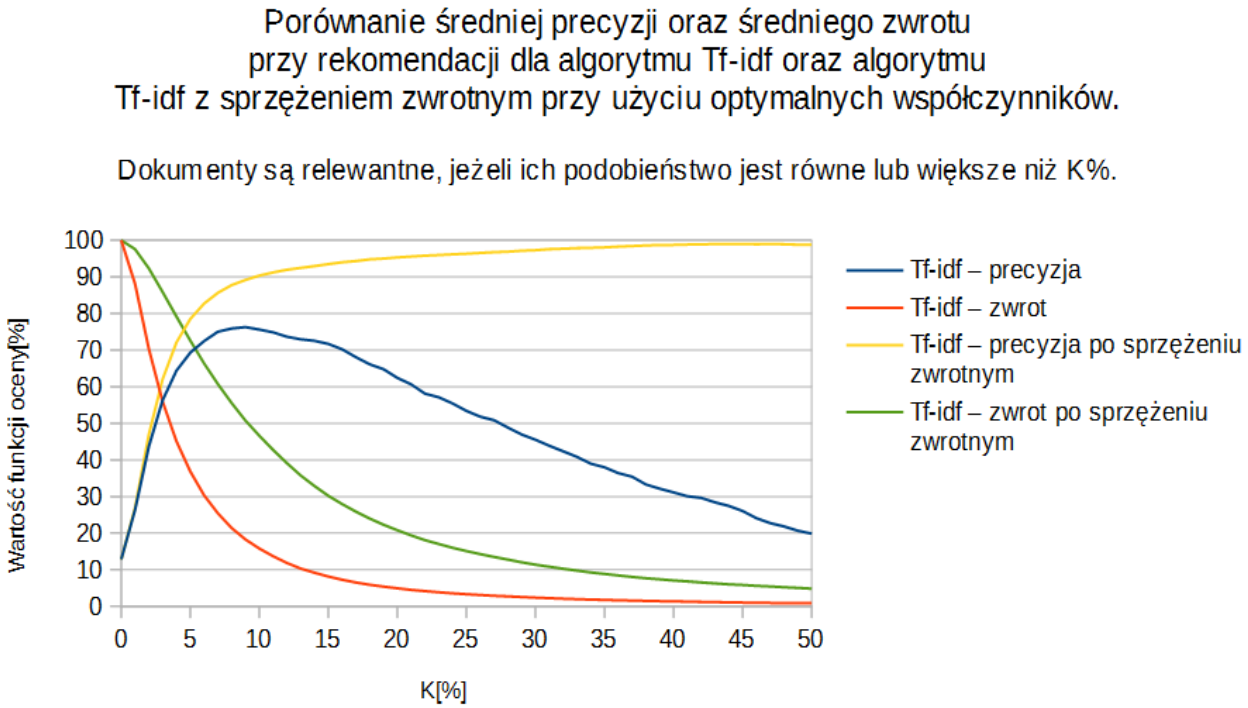
Uzyskana najlepsza kombinacja współczynników została wykorzystana do zbadania rzeczywistej wartości średniego zwrotu oraz średniej precyzji po dokonaniu sprzężenia zwrotnego przy rekomendacji dokumentów tekstowych, dla których podobieństwo dokumentów było większe lub równe niż wskazana wartość K.

Porównanie jak sprzężenie zwrotne wpływało na wartości średniego zwrotu oraz średniej precyzji w zależności od K zostało zaprezentowano na rys. 11.11 oraz 11.12, odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

Wyniki jednoznacznie wskazują, że dla obu algorytmów, zarówno wartości średniej precyzji oraz średniego zwrotu zwiększyły się po zastosowaniu sprzężenia zwrotnego z użyciem optymalnych współczynników. Należy również zwrócić szczególną uwagę na maksymalne osiągane wartości średniej precyzji (ponad 95%) dla obu algorytmów przy wykorzystaniu sprzężenia zwrotnego.



Rys. 11.11: Porównanie średniej precyzji oraz średniego zwrotu przy rekomendacji dla algorytmu zwykłego oraz algorytmu zwykłego z sprzężeniem zwrotnym przy użyciu optymalnych współczynników. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K%. Opracowanie własne.



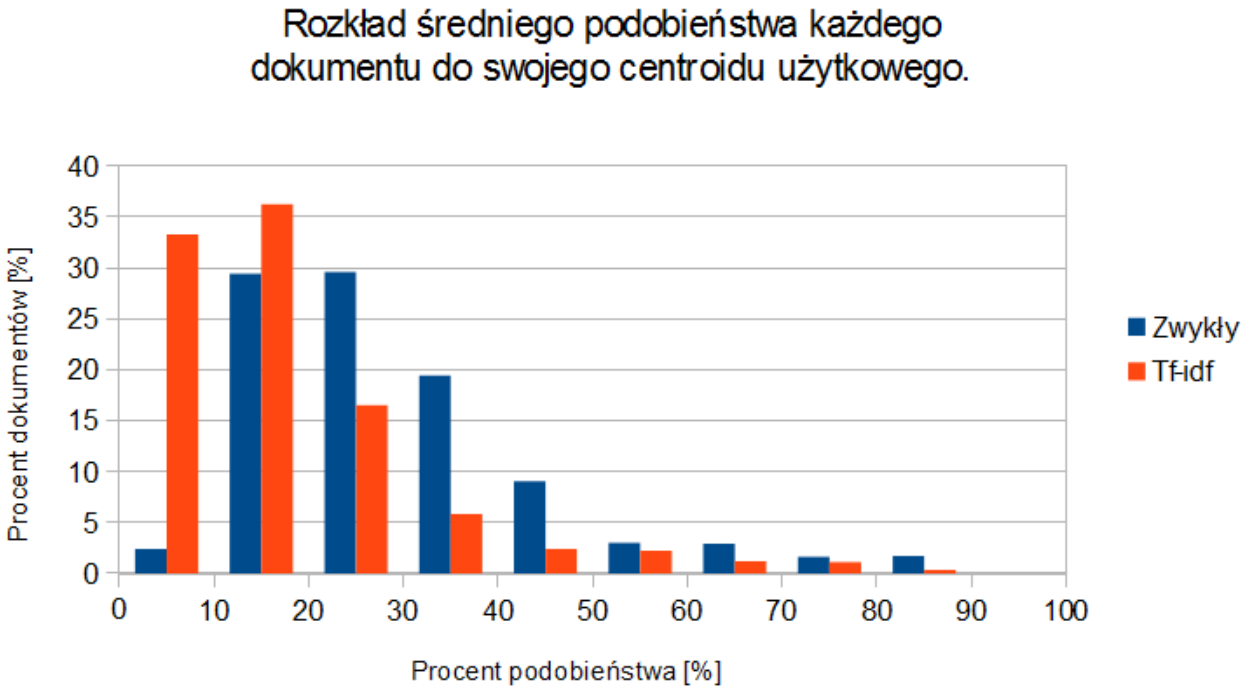
Rys. 11.12: Porównanie średniej precyzji oraz średniego zwrotu przy rekomendacji dla algorytmu Tf-idf oraz algorytmu Tf-idf z sprzężeniem zwrotnym przy użyciu optymalnych współczynników. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K%. Opracowanie własne.

Najwyższa możliwa wartość średniej F-miary dla powyższej rekomendacji z sprzężeniem zwrotnym przy użyciu optymalnych współczynników wyniosła 74,21% oraz 75,46% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.7. Wyszukiwanie w oparciu o centroidy użytkowe – klasyfikacja binarna

Następnym krokiem badań było zbadanie wpływu centroidów na uzyskiwaną średnią precyzję i średni zwrot. Centroidy pozwoliły na utworzenia reprezentanta grupy, a zatem umożliwiają łatwiejszy dostęp do informacji podobnych do centroidu. Najważniejszym zadaniem był odpowiedni wybór reprezentanta grupy. Mógł zostać on wybrany w sposób ręczny lub poprzez wybranie kilku dokumentów, które utworzyły wirtualny wektor, będący reprezentantem grupy, zgodnie z techniką tworzenia profilów metodą Rocchio.

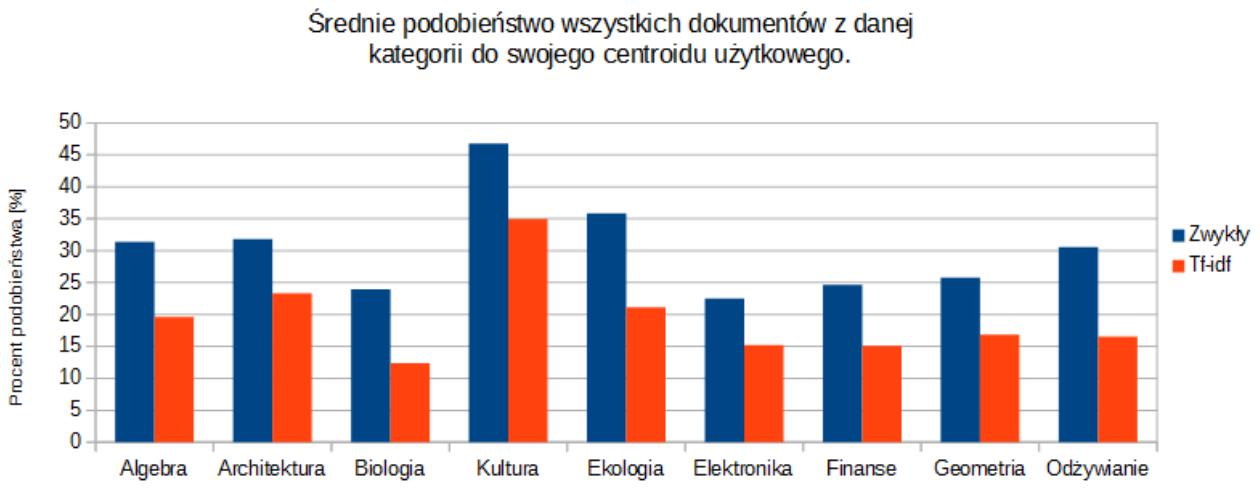
W zbiorze testowym, każda kategoria posiadała artykuł opisujący najogólniej tematykę danej kategorii, o tym samym tytule co nazwa kategorii. Do celów badawczych przyjęto, że każdy taki artykuł został centroidem użytkowym dla danej kategorii. Przykładowo, dla kategorii "Algebra" centroidem użytkowym został artykuł o tytule "Algebra". Centroidy użytkowe powinny charakteryzować się tym, że dokumenty o podobnej tematyce powinny mieć odpowiednio wysokie podobieństwo do swojego centroidu. Na rys. 11.13 zaprezentowano rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego:



Rys. 11.13: Rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego. Opracowanie własne.

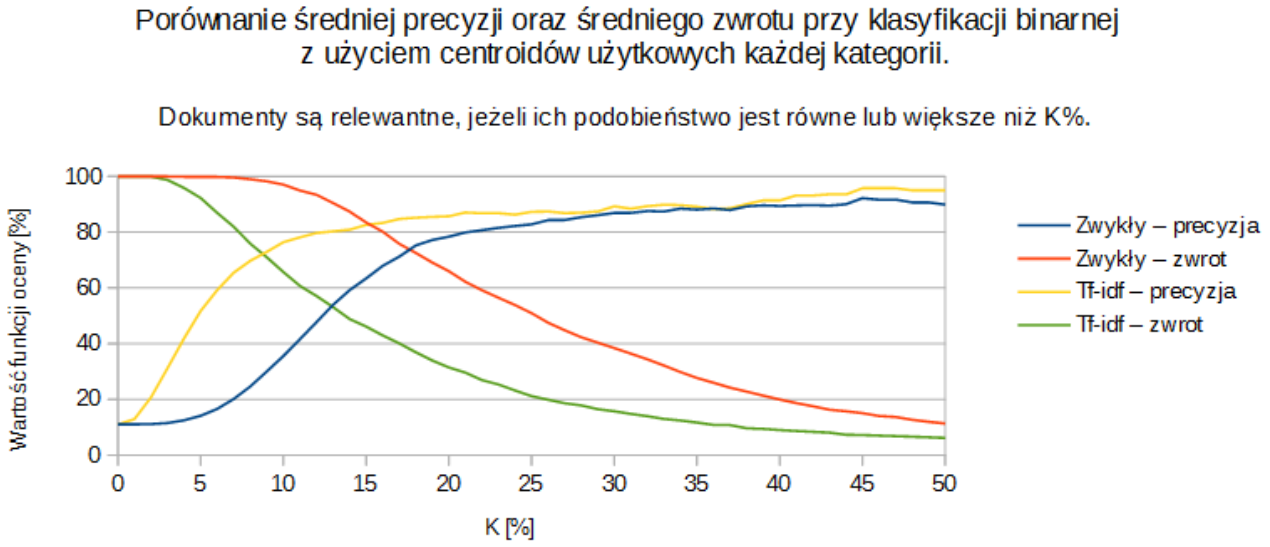
Dla techniki Tf-idf, średnie podobieństwo każdego dokumentu do swojego centroidu użytkowego wyniosło 18,46%. Dla algorytmu bez techniki Tf-idf, średnie podobieństwo wyniosło 29,8%.

Dokładniejszy podział według kategorii został zaprezentowany na rys. 11.14.



Rys. 11.14: Średnie podobieństwo wszystkich dokumentów z danej kategorii do swojego centroidu użytkowego. Opracowanie własne.

W ramach jednej poszukiwanej kategorii dokumentów, wystarczyło aby użytkownik zaproponował jeden centroid użytkowy (profil) dla danej kategorii oraz zlecił dokonanie klasyfikacji binarnej – czy każdy poszczególny dokument w kolekcji należał lub nie należał do danej kategorii. Porównanie każdego dokumentu z centroidem użytkowym przy użyciu kosinusoidalnej miary podobieństwa pozwoliło na wyznaczenie rankingu podobieństwa dla wszystkich dokumentów z całej kolekcji. Następnie należało przyjąć określony procent podobieństwa jako minimum, aby dany dokument został zaklasyfikowany do danej kategorii centroidu użytkowego. Klasyfikacja binarna pozwoliła w ten sposób na rekomendację użytkownikowi dokumentów sklasyfikowanych do danej kategorii. Również w tym przypadku funkcje precyzji oraz zwrotu pozwoliły na ocenę poprawności dokonanej klasyfikacji binarnej. Na rys. 11.15 zaprezentowano porównanie średniej precyzji oraz średniego zwrotu w zależności od przyjętego procentowego minimum podobieństwa pomiędzy dokumentem a centroidem zapewniającym klasyfikację danego dokumentu do danej kategorii.

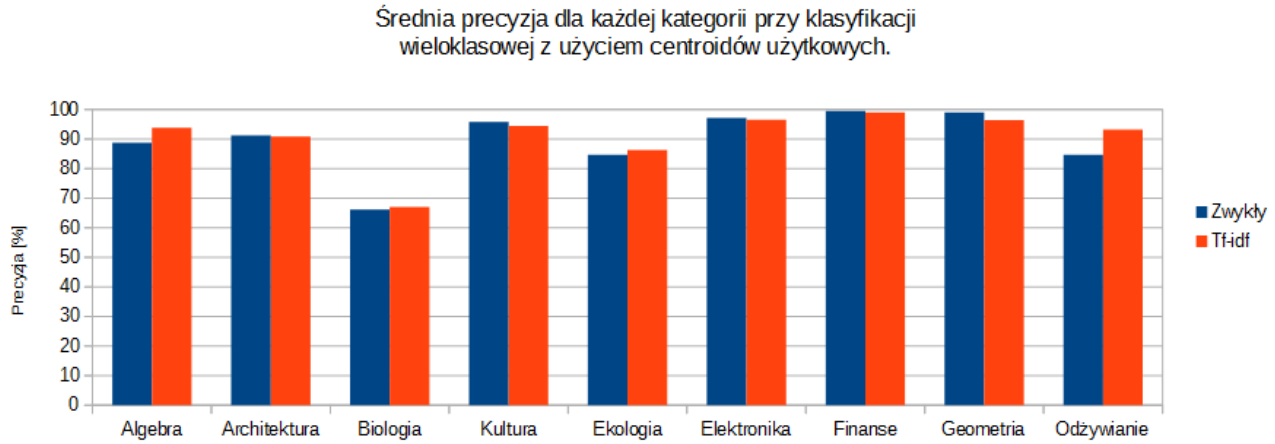


Rys. 11.15: Porównanie precyzji oraz zwrotu przy rekomendacji z użyciem centroidów użytkowych każdej kategorii. Opracowanie własne.

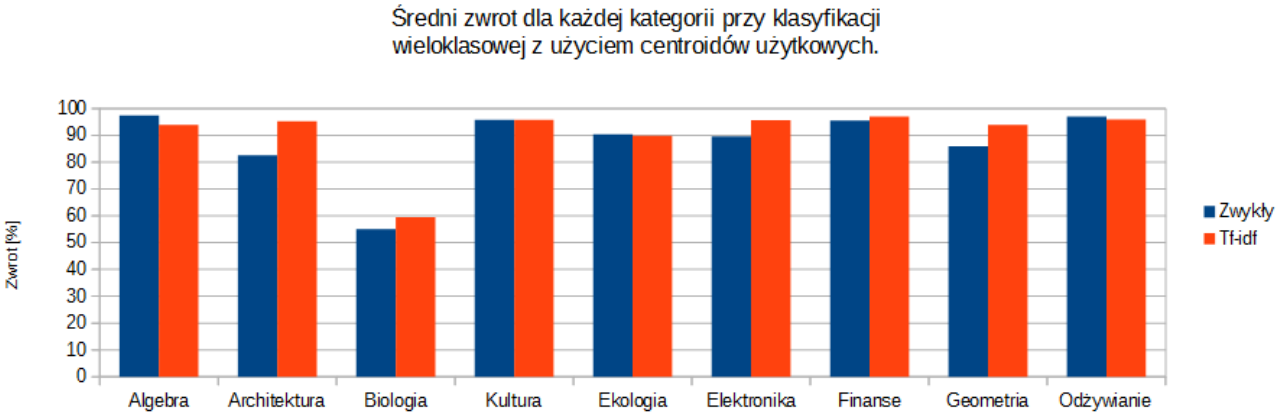
Najwyższa możliwa wartość średniej F-miary dla klasyfikacji binarnej z użyciem centroidów użytkowych każdej kategorii wyniosła 73,89% oraz 72,82% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.8. Wyszukiwanie w oparciu o centroidy użytkowe – klasyfikacja wieloklasowa

Kolejny eksperyment zbadał średnią precyzję oraz średni zwrot przy wykorzystaniu klasyfikacji wieloklasowej, wykorzystując większą wiedzę użytkownika odnośnie jego kolekcji dokumentów tekstowych. W tym przypadku należało przyjąć za założenie, że użytkownik nie tylko zdefiniował przykładowego reprezentanta poszukiwanej kategorii, lecz był również zorientowany odnośnie tego, jakie dokładnie kategorie dokumentów tekstowych można by było wyodrębnić ze swojej kolekcji. Oznaczało to, że użytkownik znając przykładowe kategorie swoich dokumentów, był w stanie wyznaczyć jednego reprezentanta grupy (centroid użytkowy) dla każdej kategorii. Następnie użytkownik mógł zlecić systemowi dokonanie klasyfikacji wieloklasowej dla wszystkich pozostałych dokumentów. Każdy dokument był porównywany z centroidem każdej kategorii przy pomocy kosinusoidalnej miary podobieństwa. Następnie każdy dokument był klasyfikowany do tej kategorii, z której centroidem użytkowym uzyskał największe podobieństwo.
Na rys. 11.16 oraz rys. 11.17 zaprezentowano średnią precyzję oraz średni zwrot dla każdej kategorii przy klasyfikacji wieloklasowej z użyciem centroidów użytkowych.



Rys. 11.16: Średnia precyzja dla każdej kategorii przy klasyfikacji wieloklasowej z użyciem centroidów użytkowych. Opracowanie własne.

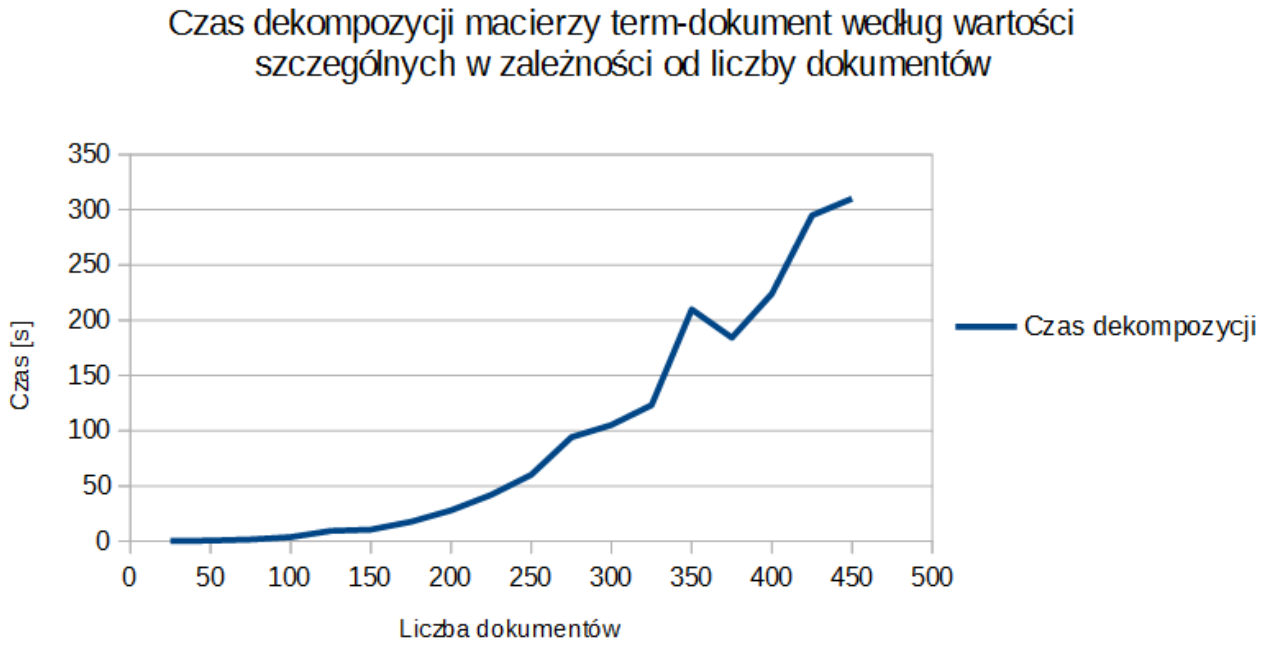


Rys. 11.17: Średni zwrot dla każdej kategorii przy klasyfikacji wieloklasowej z użyciem centroidów użytkowych. Opracowanie własne.

Średnia F-miara bez wykorzystania techniki Tf-idf dla klasyfikacji wieloklasowej z użyciem centroidów użytkowych wyniosła 89,76%, a z wykorzystaniem techniki Tf-idf 91,82%.

11.9. Optymalizacja rekomendacji z użyciem ukrytej analizy semantycznej

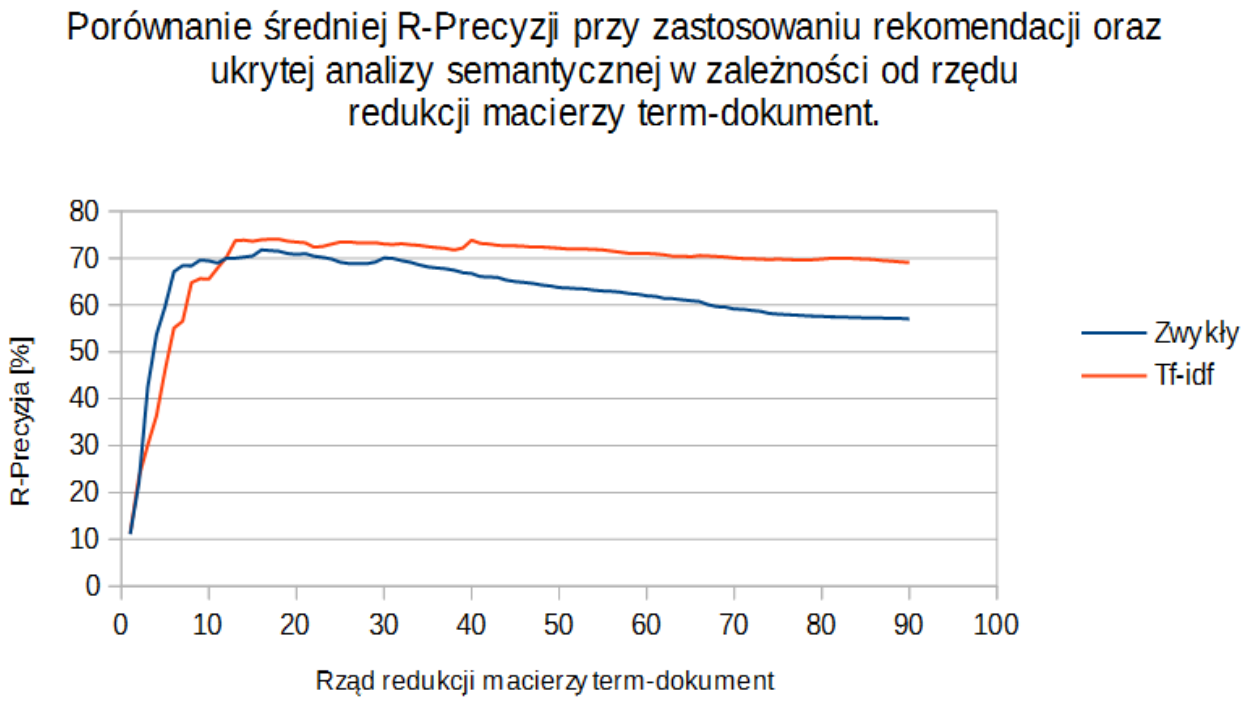
Główną problemem techniki ukrytej analizy semantycznej była złożoność obliczeniowa wynikająca z dekompozycji macierzy term-dokument według wartości szczególnych. Badania przeprowadzono z użyciem kilku programistycznych bibliotek matematycznych w języku Java, pozwalających na wykonanie rozkładu macierzy według wartości szczególnych. W badaniach wykorzystano popularne open sourcowe biblioteki JBlas, Parallel Colt, Jama oraz Commons Math. Najszybsza dekompozycja macierzy została osiągnięta przy użyciu biblioteki Commons Math, dlatego została ona wybrana do dalszych badań.
Na rys. 11.18 zaprezentowano czas dekompozycji macierzy term-dokument według wartości szczególnych w zależności od liczby dokumentów.



Rys. 11.18: Czas dekompozycji macierzy term-dokument według wartości szczególnych w zależności od liczby dokumentów. Opracowanie własne.

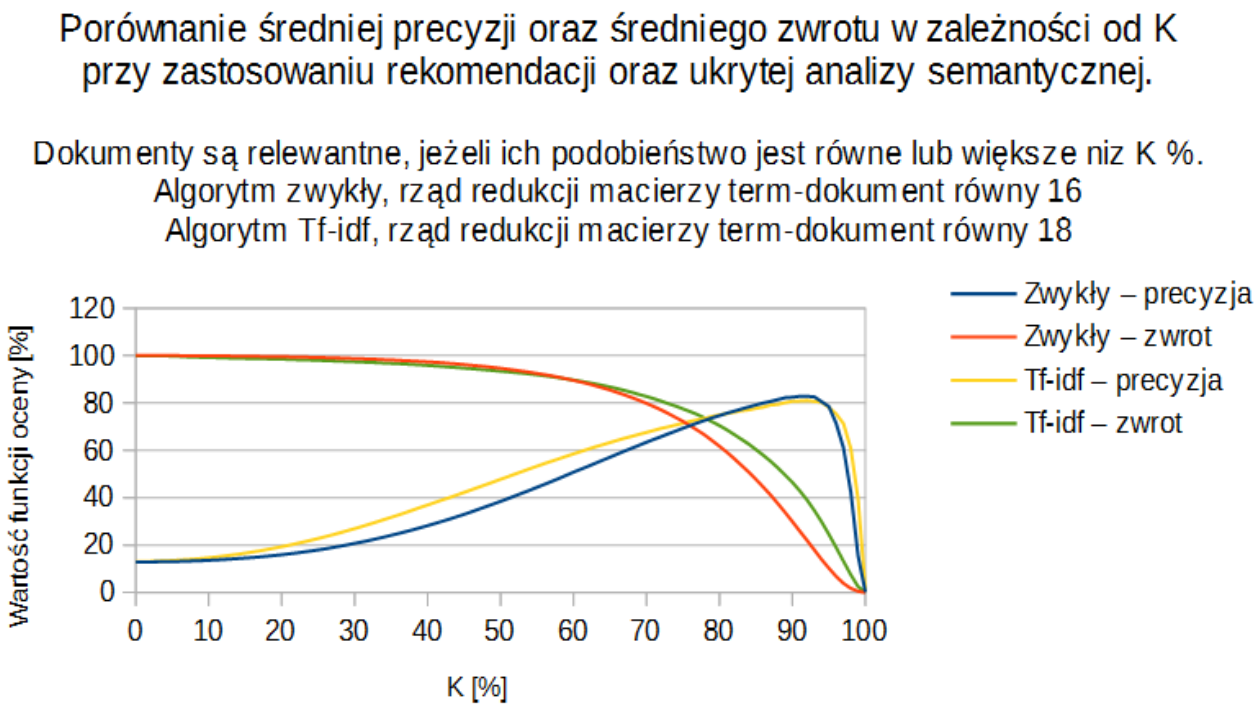
Na rys. 11.18 można zaobserwować, że czas dekompozycji rósł wykładniczo w zależności od ilości dokumentów. Dodatkowo, wraz z wzrostem liczby dokumentów, rosła też łączna liczba możliwych termów w macierzy term-dokument. Dla całej kolekcji testowej, liczącej 1162 dokumentów, czas dekompozycji wynosił 140 minut. Dla porównania, średni czas wyznaczenia standardowego rankingu podobieństwa jednego dokumentu do wszystkich pozostałych dokumentów z powyższej kolekcji z użyciem kosinusoidalnej miary podobieństwa wynosił około 150 milisekund. Z tego wynika, że ukryta analiza semantyczna powinna być wykonywana tylko jednorazowo, a jej wyniki zapamiętane przez system, aby użytkownik mógł później korzystać z zredukowanej macierzy term-dokument do szybkiego wyznaczania podobieństwa pomiędzy dokumentami.

Jednym z celów badania dla ukrytej analizy semantycznej było wyznaczenie optymalnego rzędu redukcji macierzy term-dokument. Dokonano redukcji macierzy term-dokument składającej się z całej kolekcji dokumentów testowych i następnie korzystając z ukrytej analizy semantycznej dokonano rekomendacji dokumentów relewantnych dla każdego dokumentu ze zbioru testowego. W pierwszym etapie badań dokonano hipotetycznego założenia, że algorytm posiada wiedzę, jak wielki ilościowo zbiór informacji relewantnych dla każdego dokumentu powinien zaproponować. W ten sposób, porównując średnią R-precyzję, wyliczono najlepszy rząd redukcji macierzy term-dokument, który został później użyty do realnej rekomendacji dokumentów tekstowych bez żadnych założeń.



Rys. 11.19: Porównanie średniej R-Precyzji przy zastosowaniu rekomendacji oraz ukrytej analizy semantycznej w zależności od rzędu redukcji macierzy term-dokument. Opracowanie własne.

Na rys. 11.19 można zaobserwować, że średnia R-precyzja malała dla rzędów redukcji macierzy term-dokument mniejszych niż 8. Przyczyną tego faktu było to, że przedstawienie dokumentów tekstowych w reprezentacji mniejszej niż 8-wymiarowej powodowało zbyt dużą utratę informacji o dokumentach tekstowych, uniemożliwiając skuteczne ich porównywanie.
Najwyższą wartość średniej R-precyzji (71,78%) bez wykorzystania techniki Tf-idf uzyskano dla rzędu redukcji macierzy term-dokument o wartości 16. Wykorzystując technikę Tf-idf, najwyższą wartość średniej R-precyzji (74,03%) uzyskano dla rzędu redukcji macierzy term-dokument o wartości 18. Uzyskane wartości rzędu redukcji macierzy term-dokument zostały wykorzystane do zbadania rzeczywistej wartości średniego zwrotu oraz średniej precyzji przy rekomendacji dokumentów tekstowych, dla których podobieństwo dokumentów było większe lub równe niż wskazana wartość K (rys. 11.20).



Rys. 11.20: Porównanie średniej precyzji oraz średniego zwrotu w zależności od K przy zastosowaniu rekomendacji oraz ukrytej analizy semantycznej. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K %. Rząd redukcji macierzy term-dokument równy 16 oraz 18 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.

Przebieg wartości średniego zwrotu oraz średniej precyzji zarówno dla obu algorytmów z wykorzystaniem ukrytej analizy semantycznej różnił się bardzo od algorytmów, które nie skorzystały z techniki ukrytej analizy semantycznej (rys. 11.6). Na rys. 11.20 można zaobserwować, że najwyższe wartości średniej precyzji, 82,83% dla algorytmu zwykłego oraz 81,02% dla algorytmu Tf-idf, zostały osiągnięte dla K równego odpowiednio 91 oraz 92. Oznacza to, że redukcja macierzy term-dokument rzeczywiście wyostrzyła zależności pomiędzy dokumentami tekstowymi i pozwoliła na zgrupowanie dokumentów o podobnej tematyce przy pomocy reprezentacji dokumentów tekstowych o mniejszej ilości wymiarów. Dla porównania, nie korzystając z ukrytej analizy semantycznej, największa możliwa do uzyskania wartość średniej precyzji wyniosła 69,80% oraz 75,61% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf (rys. 11.6).

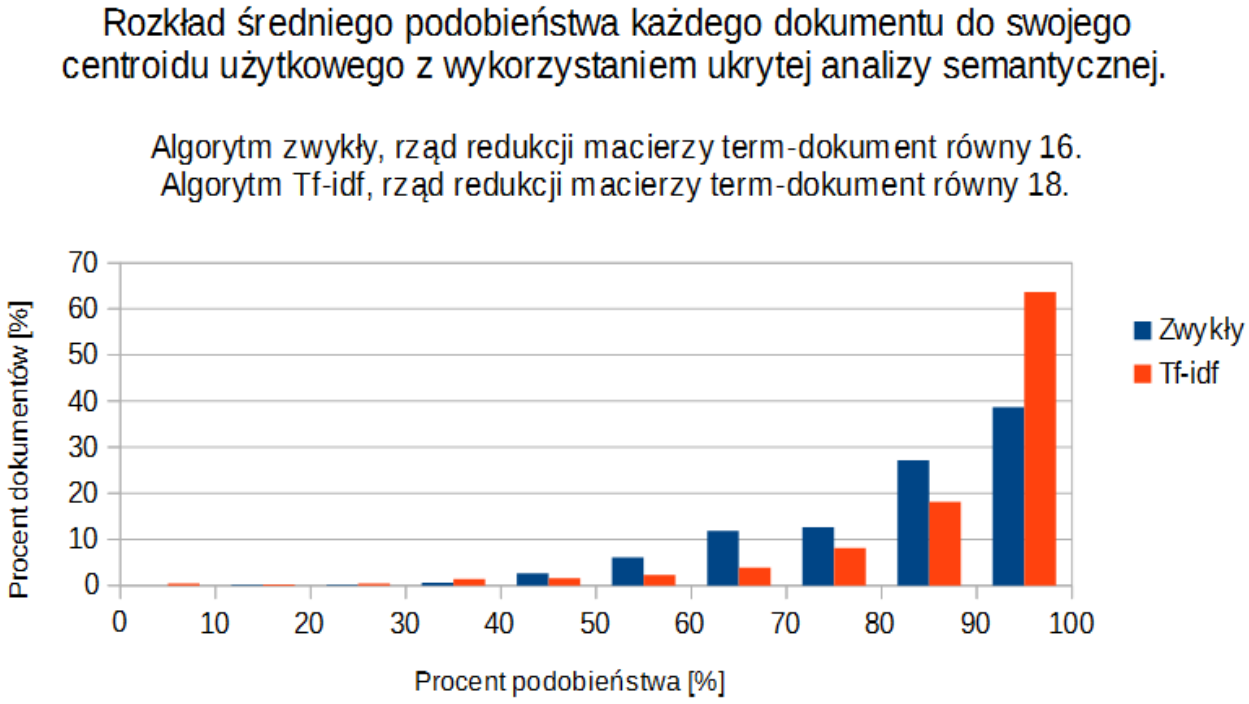
Należy również zauważyć, że dla K równego 92, funkcje precyzji osiągnęły swoje ekstremum (maksimum globalne). Była to wartość graniczna, powyżej którego wartości średniej precyzji dla obu algorytmów zaczynały drastycznie maleć, ponieważ algorytm nie był w stanie zwrócić odpowiedniej ilości dokumentów, mających tak wysokie podobieństwo do wskazanego dokumentu.

Najwyższa możliwa wartość średniej F-miary dla rekomendacji z użyciem ukrytej analizy semantycznej wyniosła 70,99% oraz 74,58% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.10. Wyszukiwanie w oparciu o centroidy użytkowe z wykorzystaniem ukrytej analizy semantycznej – klasyfikacja binarna

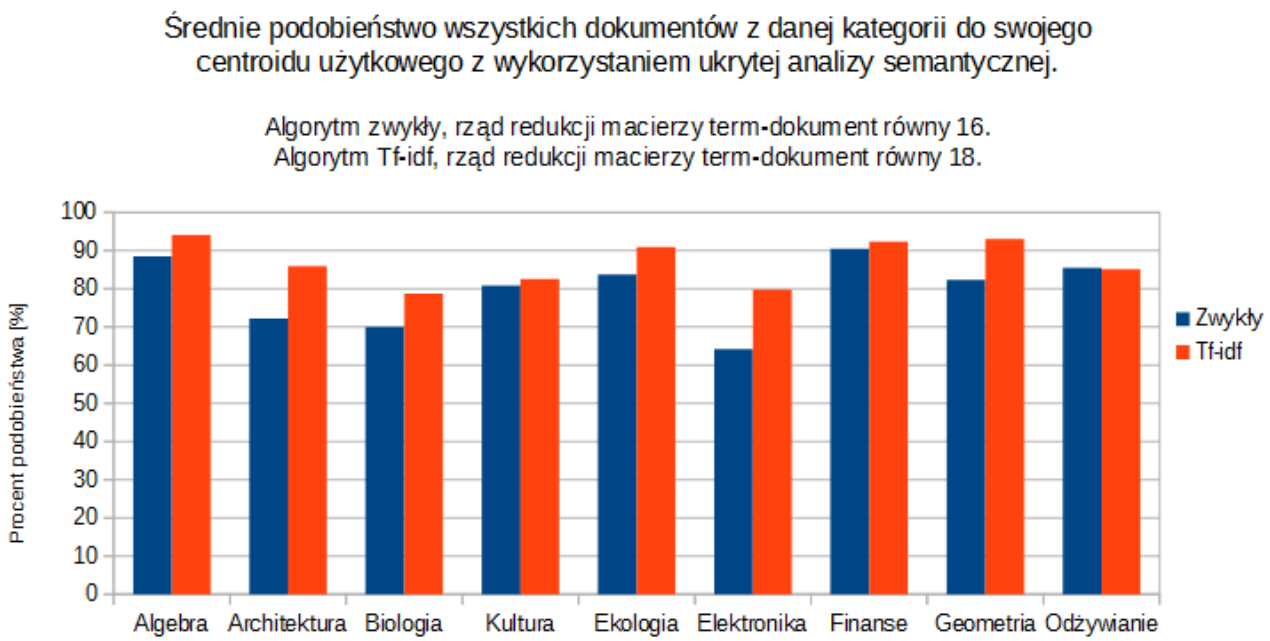
Następnym krokiem badań było zbadanie wpływu centroidów użytkowych na uzyskiwaną średnią precyzję i średni zwrot przy wykorzystywaniu ukrytej analizy semantycznej. Schemat postępowania klasyfikacji binarnej z wykorzystaniem ukrytej analizy semantycznej był identyczny jak w podrozdziale 11.7, jednak wpierw została również dokonana redukcja macierzy term-dokument do macierzy dużo niższego rzędu.

Na rys. 11.21 zaprezentowano rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego z wykorzystaniem ukrytej analizy semantycznej (rząd redukcji został wybrany na podstawie wyników badań z podrozdziału 11.9):



Rys. 11.21: Rozkład średniego podobieństwa każdego dokumentu do swojego centroidu użytkowego z wykorzystaniem ukrytej analizy semantycznej. Rząd redukcji macierzy term-dokument równy 16 oraz 18 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.

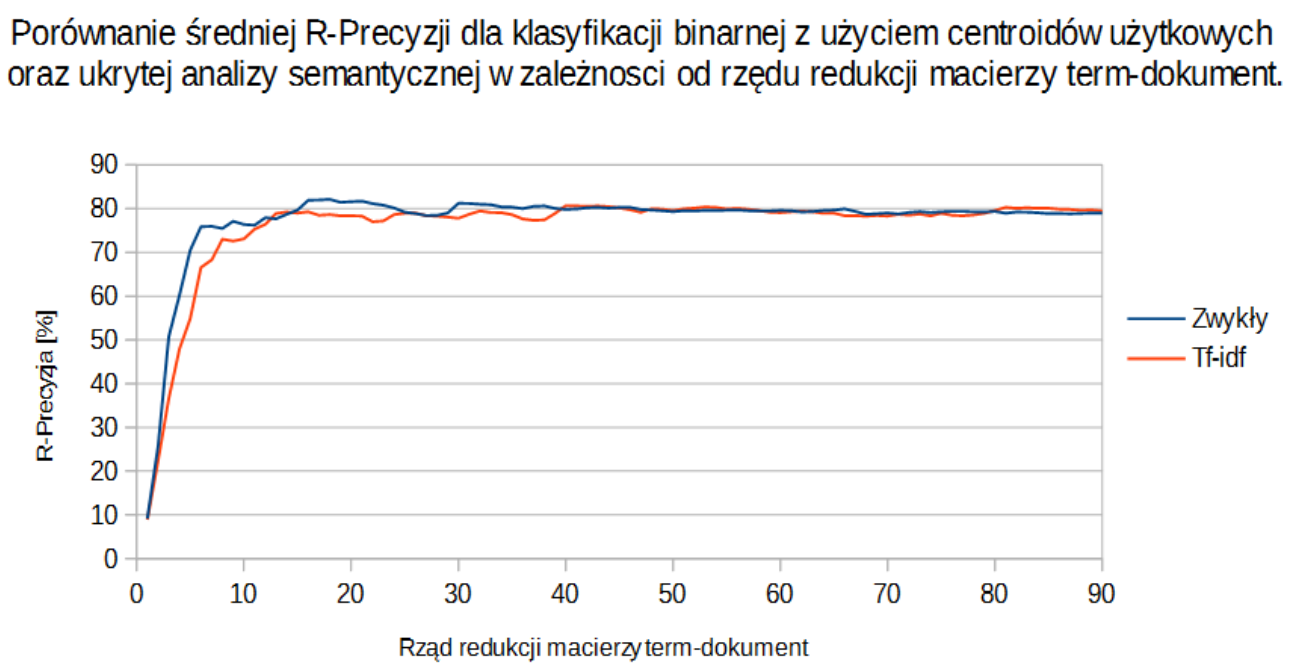
Dla techniki Tf-idf, średnie podobieństwo każdego dokumentu do swojego centroidu użytkowego wyniosło 87,89%. Dla algorytmu bez techniki Tf-idf, średnie podobieństwo wyniosło 81,75%. Dokładniejszy podział na kategorie został zaprezentowany na rys. 11.20.



Rys. 11.22: Średnie podobieństwo wszystkich dokumentów z danej kategorii do swojego centroidu użytkowego z wykorzystaniem ukrytej analizy semantycznej. Rząd redukcji macierzy term-dokument równy 16 oraz 18 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.

Wyniki badań zaprezentowane na rys. 11.21 oraz rys. 11.22 wykazują, że redukcja macierzy term-dokument pozwoliła na znaczące zwiększenie podobieństwa dokumentów o podobnej tematyce do swojego centroidu użytkowego. Dla obu algorytmów, największy procent dokumentów posiadał podobieństwo do swojego centroidu użytkowego w przedziale 90-100% Dla porównania, nie wykorzystując ukrytej analizy semantycznej, największy procent dokumentów posiadał podobieństwo do swojego centroidu użytkowego w przedziale 0-20% oraz 10-30% odpowiednio dla algorytmu Tf-idf oraz algorytmu zwykłego (rys. 11.13).

Podobnie jak w podrozdziale 11.7, lecz korzystając w tym przypadku także z ukrytej analizy semantycznej, porównanie każdego dokumentu z centroidem użytkowym przy użyciu kosinusoidalnej miary podobieństwa pozwoliło na wyznaczenie rankingu podobieństwa dla wszystkich dokumentów z całej kolekcji. Również w tym etapie badań, ponownie dokonano hipotetycznego założenia, że algorytm posiada wiedzę, jak wielki ilościowo zbiór informacji relewantnych dla każdego dokumentu powinien zaproponować. W ten sposób, porównując średnią R-precyzję, wyliczono najlepszy rząd redukcji macierzy term-dokument (rys. 11.23), który został później użyty do realnej klasyfikacji binarnej dokumentów tekstowych bez żadnych założeń.

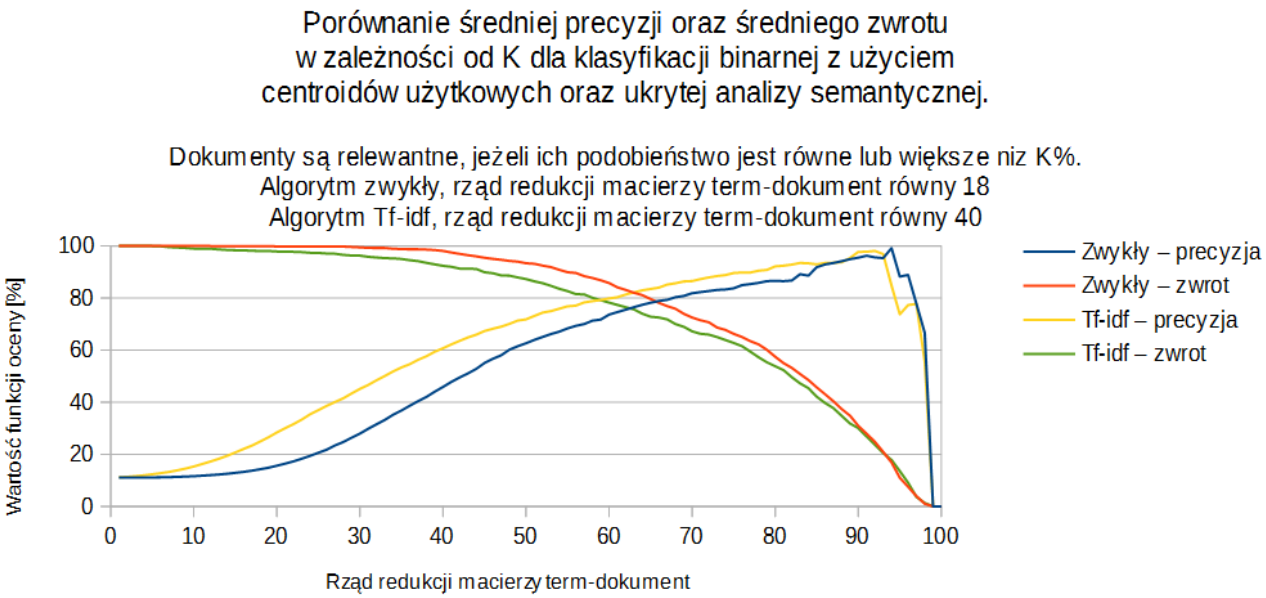


Rys. 11.23: Porównanie średniej R-Precyzji dla klasyfikacji binarnej z użyciem centroidów użytkowych oraz ukrytej analizy semantycznej w zależności od rzędu redukcji macierzy term-dokument. Opracowanie własne.

Na rys. 11.23 można ponownie zaobserwować, że średnia R-precyzja malała dla rzędów redukcji macierzy term-dokument mniejszych niż 8. Ponadto, dla obu algorytmów od pewnego rzędu redukcji macierzy term-dokument, średnia R-precyzja zaczynała stabilizować się na poziomie około 80%. Oznacza to, że niepotrzebne było stosowanie dużo większych wymiarowo reprezentacji dokumentów tekstowych, gdyż nie wpłynęło to znacząco na jakość wyszukiwania informacji. Ponadto macierze o mniejszej ilości wymiarów potrzebowały mniejszej ilości zasobów sprzętowych do ich przetwarzania.

Najwyższą wartość średniej R-precyzji (82,12%) dla klasyfikacji binarnej bez wykorzystania techniki Tf-idf uzyskano dla rzędu redukcji macierzy term-dokument o wartości 18. Wykorzystując technikę Tf-idf, najwyższą wartość średniej R-precyzji (80,68%) dla klasyfikacji binarnej uzyskano dla rzędu redukcji macierzy term-dokument o wartości 40. Uzyskane wartości rzędu redukcji macierzy term-dokument zostały wykorzystane do zbadania rzeczywistej wartości średniego zwrotu oraz średniej precyzji przy klasyfikacji binarnej dokumentów tekstowych, dla których podobieństwo dokumentów było większe lub równe niż wskazana wartość K.

Na rys. 11.24 zaprezentowano porównanie średniej precyzji oraz średniego zwrotu dla klasyfikacji binarnej w zależności od przyjętego procentowego minimum podobieństwa pomiędzy dokumentem a centroidem zapewniającym klasyfikację danego dokumentu do danej kategorii.



Rys. 11.24: Porównanie średniej precyzji oraz średniego zwrotu w zależności od K przy zastosowaniu klasyfikacji binarnej oraz ukrytej analizy semantycznej. Dokumenty są relewantne, jeżeli ich podobieństwo jest równe lub większe niż K %. Rząd redukcji macierzy term-dokument równy 18 oraz 40 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.

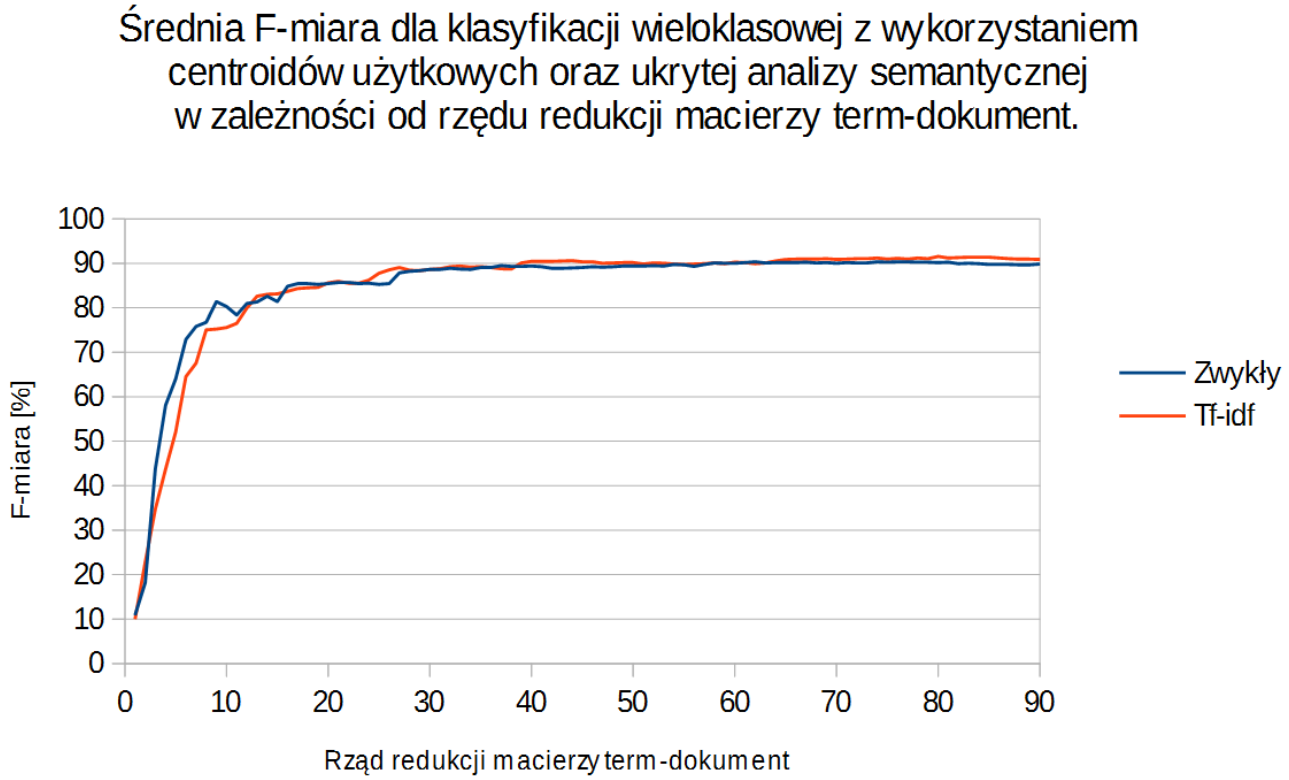
Przebieg wartości średniej precyzji oraz średniego zwrotu dla klasyfikacji binarnej z wykorzystaniem ukrytej analizy semantycznej zaprezentowany na rys. 11.24 był bardzo zbliżony do wyników zaprezentowanych na rys. 11.20. Jednak w tym przypadku należy zwrócić szczególną uwagę na maksymalne osiągane wartości średniej precyzji. Korzystając z ukrytej analizy semantycznej, najwyższe wartości średniej precyzji – 99,17% dla algorytmu zwykłego oraz 98,10% dla algorytmu Tf-idf, zostały osiągnięte dla K równego odpowiednio 94 oraz 92.

Najwyższa możliwa wartość średniej F-miary dla klasyfikacji binarnej z użyciem centroidów użytkowych każdej kategorii oraz ukrytej analizy semantycznej wyniosła 79,26% oraz 79,83% odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf.

11.11. Wyszukiwanie w oparciu o centroidy użytkowe z wykorzystaniem ukrytej analizy semantycznej – klasyfikacja wieloklasowa

Ostatnim etapem badań było wykonanie klasyfikacji wieloklasowej z wykorzystaniem ukrytej analizy semantycznej. Schemat postępowania klasyfikacji wieloklasowej z wykorzystaniem ukrytej analizy semantycznej był identyczny jak w podrozdziale 11.8, jednak wpierw została również dokonana redukcja macierzy term-dokument do macierzy dużo niższego rzędu.

W przypadku klasyfikacji wieloklasowej dodatkowym parametrem konfiguracyjnym przy użyciu ukrytej analizy semantycznej był rząd redukcji macierzy term-dokument. W tym celu przeprowadzono badanie, w którym zbadano wartość średniej F-miary dla klasyfikacji wieloklasowej z wykorzystaniem centroidów użytkowych oraz ukrytej analizy semantycznej w zależności od rzędu redukcji macierzy term-dokument. Wyniki powyższego badania zaprezentowano na rys. 11.25.

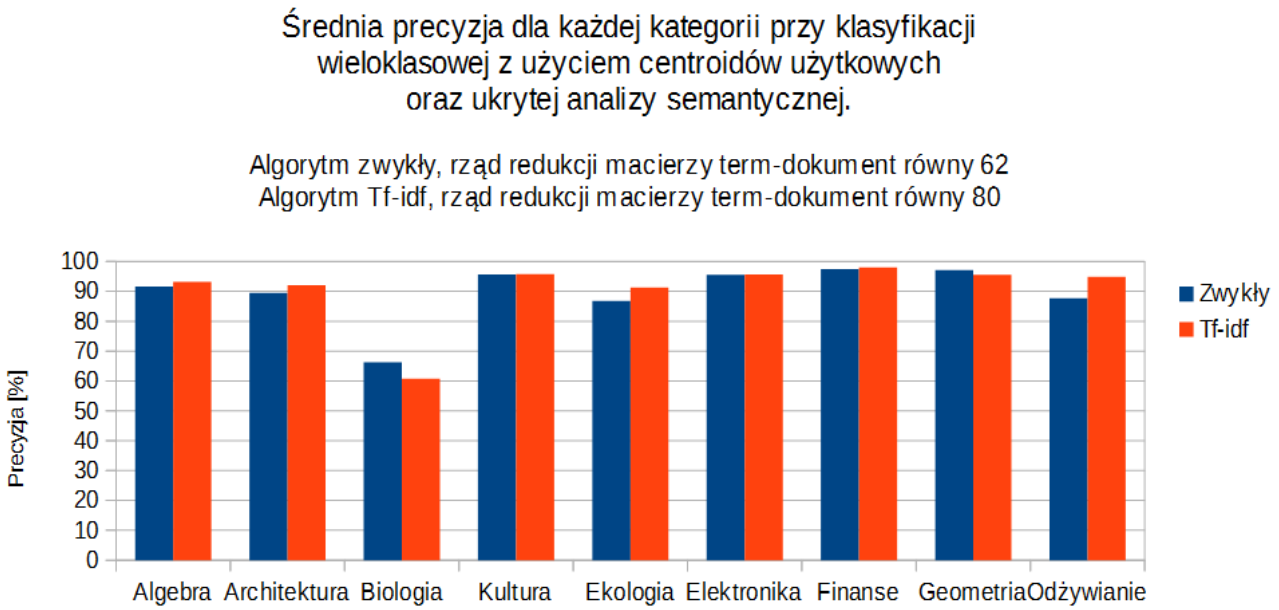


Rys. 11.25: Średnia F-miara dla klasyfikacji wieloklasowej z wykorzystaniem centroidów użytkowych oraz ukrytej analizy semantycznej w zależności od rzędu redukcji macierzy term-dokument. Opracowanie własne.

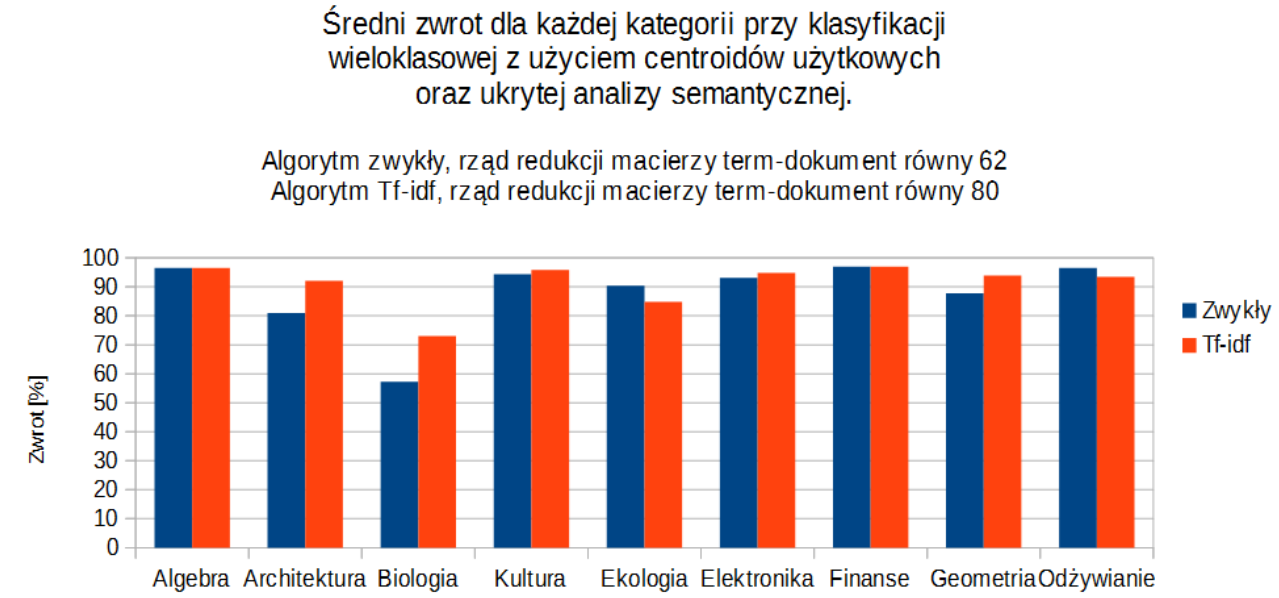
Na rys. 11.25 można zaobserwować, że dla obu algorytmów od pewnego rzędu redukcji macierzy term-dokument, średnia F-miara stabilizowała się na poziomie około 90%. Również w tym przypadku oznacza to, że niepotrzebne było stosowanie dużo większych wymiarowo reprezentacji dokumentów tekstowych, gdyż nie wpłynęło to znacząco na jakość wyszukiwania informacji.

Najwyższą wartość średniej F-miary (90,36%) dla klasyfikacji wieloklasowej bez wykorzystania techniki Tf-idf uzyskano dla rzędu redukcji macierzy term-dokument o wartości 62. Wykorzystując technikę Tf-idf, najwyższą wartość średniej F-miary (91,57%) dla klasyfikacji wieloklasowej uzyskano dla rzędu redukcji macierzy term-dokument o wartości 80.

Dla powyższych wartości rzędu redukcji macierzy term-dokument zostały zaprezentowane też wartości średniej precyzji oraz średniego zwrotu dla każdej z kategorii. Wyniki zostały zaprezentowane na rys. 11.26 oraz rys. 11.27.



Rys. 11.26: Średnia precyzja dla każdej kategorii przy klasyfikacji wieloklasowej z użyciem centroidów użytkowych oraz ukrytej analizy semantycznej. Rząd redukcji macierzy term-dokument równy 18 oraz 40 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.



Rys. 11.27: Średni zwrot dla każdej kategorii przy klasyfikacji wieloklasowej z użyciem centroidów użytkowych oraz ukrytej analizy semantycznej. Rząd redukcji macierzy term-dokument równy 18 oraz 40 odpowiednio dla algorytmu zwykłego oraz Tf-idf. Opracowanie własne.

Szczególną uwagę należy zwrócić na średni zwrot uzyskiwany przez kategorię „Biologia", która osiągała najniższe wyniki dla wszystkich poprzednich badań. Po zastosowaniu techniki Tf-idf oraz ukrytej analizy semantycznej, wartość średniego zwrotu dla kategorii „Biologia" wyniosła 73,03%. Dla porównania, nie korzystając z techniki ukrytej analizy semantycznej oraz techniki Tf-idf, wartość średniego zwrotu dla powyższej kategorii wyniosła 55,06% (rys. 11.17).

12. Podsumowanie

W 1969 roku powstanie Internetu zrewolucjonizowało oblicze świata, co spowodowało, że świat wkroczył w nową erę globalizacji, pozwalają na szybką wymianę informacji. Coraz to większa liczba osób podłączonych do internetu pozwoliła na generowanie oraz przetwarzanie coraz to większej ilości danych. Nowoczesne narzędzia komunikacji, aplikacje i społeczności internetowe powodują, że możliwe jest wysyłanie i odbieranie informacji bez względu na czas i miejsce. Era cyfryzacji pozwoliła na łatwiejszą wymianę informacji oraz poznawanie opinii innych ludzi, jednak spowodowała również problem przeciążenia informacyjnego, który jest obecnie naturalną konsekwencją rosnącej podaży informacji. Z powodu przeciążenia informacyjnego, problemem staję się dotarcie do najbardziej potrzebnych informacji w celu skorzystania z nich.

Osiągnięcia z dziedziny wyszukiwania informacji, mającej już ponad 50 lat aktywnej działalności, doprowadziły do powstania wiele modelów służących odpowiedniemu wyszukiwaniu informacji dla zbioru dokumentów tekstowych. Pierwsze systemy wyszukiwania informacji, na długo jeszcze przed powstaniem internetu, powstały na potrzeby środowiska naukowego i służyły wąskiemu gronu użytkowników, którzy mieli bardzo konkretnie sprecyzowane potrzeby informacyjne. W 1945 Vannevar Bush opublikował artykuł “As We May Think”, który zaprezentował ideę automatycznego dostępu do dużej ilości zgromadzonej wiedzy [25]. W latach 50 i 60 XX wieku stworzono pierwsze systemy do automatycznego wyszukiwania informacji z użyciem komputera osobistego. W krótkim czasie dziedzina wyszukiwania informacji z akademickiej dyscypliny stała się fundamentem do rozwoju idei szybkiego dostępu do informacji, również dla osób niezwiązanych z środowiskiem naukowym.

Wraz z rozwojem osiągnięć z dziedziny wyszukiwania informacji, pojawiła się możliwość stworzenia systemów do zarządzania informacją osobistą, tzw. personalnych elektronicznych organizerów, których podstawowymi funkcjonalnością są zarządzanie kontaktami, wydarzeniami, spotkaniami, zadaniami czy innymi ważnymi dla użytkownika notatkami. Systemy do zarządzania informacją osobistą muszą najczęściej indeksować dane częściowo strukturyzowane, czyli w głównej mierze dane tekstowe. Głównym problemem systemów wyszukiwawczych operujących na danych tekstowych jest zdefiniowanie funkcji, która pozwoli w sposób jak najbardziej zbliżony do potrzeb użytkownika, na określenie relewantności danego dokumentu tekstowego dla danego zapytania. Z powodu złożoności obecnych rozwiązań, niezbędne było przeprowadzenie serii eksperymentów oraz wypracowanie wspólnego kryterium oceny, który pozwoliło na ewaluację różnych rozwiązań pod względem skuteczności wyszukiwania informacji relewantnych dla użytkownika. Głównym celem pracy było zaimplementowanie oraz porównanie różnych strategii wyszukiwania informacji z użyciem modelu przestrzeni wektorowej, cel ten udało się w pełni zrealizować.

W ramach części teoretycznej pracy omówiono podstawowe aspekty związane z informacją, kategoryzacją informacji oraz systemami do zarządzania informacją osobistą. Następnie zdefiniowano podstawowe problemy informacyjne w zarządzaniu informacją osobistą oraz dokonano prezentacji podstawowych pojęć i klasyfikacji związanej z zagadnieniami wyszukiwania i rekomendacji informacji. Omówiono również szczegółowo sposób tworzenia reprezentacji wektorowej dokumentu oraz zaprezentowano różnice wynikające z normalizacji dokumentów tekstowych z użyciem algorytmu stemmingu Snowball oraz słownika morfologicznego Ispell. W ramach przeglądu zaprezentowano kilka modeli wyszukiwania informacji. Szczegółowej analizie zostały poddane model boolowski oraz model przestrzeni wektorowej.

W ramach zagadnienia optymalizacji wyników wyszukiwania, opisano sposób użycia słowników oraz technikę Tf-idf, pozwalającą na odpowiednie zbilansowanie wag termów na podstawie ich wartości dyskryminacyjnej. Zaprezentowano również metodę sprzężenia zwrotnego oraz szczegółowo omówiono technikę ukrytej analizy semantycznej, wraz z zaprezentowaniem geometrycznej reprezentacji dokumentów przy użyciu zredukowanej macierzy term-dokument. Zaprezentowano również podstawową systematykę metod klasyfikacji oraz omówiono technikę klasyfikacji dokumentów tekstowych z użyciem profilów metodą Rocchio.

W części badawczej zaimplementowano podstawową strategię rekomendacji z użyciem kosinusoidalnej miary podobieństwa oraz opracowano i zaimplementowano własną strategię wyszukiwania informacji na podstawie najczęściej występujących termów dla każdej pary dokumentów. Do badań testowych niezbędne było zebranie przykładowych dokumentów testowych oraz przetworzenie ich do reprezentacji wektorowej. Każdy dokument należał do jednej kategorii, ponadto każda kategoria posiadała przykładowego reprezentanta kategorii (centroid użytkowy) w postaci konkretnego dokumentu o takim samym tytule jak nazwa kategorii.

W ramach każdego zapytania dla każdej zaimplementowanej strategii, celem było zwrócenie jak największej ilości dokumentów relewantnych (rekomendacji prawdziwie pozytywnych), z możliwie jak najmniejszą ilością dokumentów nierelewantnych (rekomendacji fałszywie pozytywnych). Ponieważ funkcje precyzji i zwrotu konkurują ze sobą, dlatego do końcowego porównania różnych strategii użyto F-miary (średniej harmonicznej precyzji i zwrotu) jako wspólnego kryterium oceny.

Każda strategia posiadała odpowiednie parametry konfiguracyjne, które wpłynęły na wyniki uzyskiwane przez daną strategię. Końcowe badania były poprzedzone eksperymentem, w którym zbadano 2 techniki normalizacji oraz ich wpływ na skuteczność wyszukiwania informacji. Badania wykazały, że zarówno technika stemmingu jak i lematyzacji, osiągnęła podobne wyniki, dlatego do końcowego porównania strategii użyto tylko techniki stemmingu.

Z przeprowadzonych badań wynika, że najlepsze wyniki zostały osiągnięte dla klasyfikacji wieloklasowej z wykorzystaniem techniki Tf-idf (91,82%). Klasyfikacja wieloklasowa, w porównaniu do klasyfikacji binarnej oraz rekomendacji, okazała się najskuteczniejszym sposobem wyszukiwania informacji. Jednak technika ta była oparta na mało realistycznym założeniu, że użytkownik był w stanie wyznaczyć dokładnie jednego reprezentanta grupy (centroid użytkowy) dla każdej kategorii.

Badania wykazały, że ukryta analiza semantyczna znacząco poprawiła wyniki dla rekomendacji z użyciem kosinusoidalnej miary podobieństwa oraz klasyfikacji binarnej. Jednak dla klasyfikacji wieloklasowej, wyniki z użyciem ukrytej analizy semantycznej oraz jej braku były już porównywalne.

Klasyfikacja binarna opierała się na dużo bardziej realistycznym założeniu w porównaniu do klasyfikacji wieloklasowej. W przypadku klasyfikacji binarnej, użytkownik zdefiniował przykładowego reprezentanta kategorii dokumentów, których poszukiwał. Zadanie te można było by porównać do wyszukiwania w oparciu o dokładnie zdefiniowany zbiór słów kluczowych, jednak w tym przypadku zapytaniem był konkretny dokument, wybrany przez użytkownika jako centroid użytkowy. Należy uwzględnić przy tym fakt, że dla techniki Tf-idf średnie podobieństwo każdego dokumentu do swojego centroidu użytkowego wyniosło tylko 18,46%. Dla algorytmu bez techniki Tf-idf, średnie podobieństwo wyniosło 29,8%. Po zastosowaniu ukrytej analizy semantycznej, dla techniki Tf-idf średnie podobieństwo każdego dokumentu do swojego centroidu użytkowego wyniosło 87,89%, a dla algorytmu bez techniki Tf-idf średnie podobieństwo wyniosło 81,75%.

Końcowy wynik dla klasyfikacji binarnej z użyciem ukrytej analizy semantycznej wyniósł 79,26% oraz 79,83%, odpowiednio dla algorytmu zwykłego oraz algorytmu Tf-idf. Zastosowanie ukrytej analizy semantycznej pozwoliło zatem na znaczącą poprawę wyników wyszukiwania dla klasyfikacji binarnej, opierającej się na realistycznych założeniach, dlatego należy uznać, że wynik tego badania był najistotniejszy dla całej pracy.

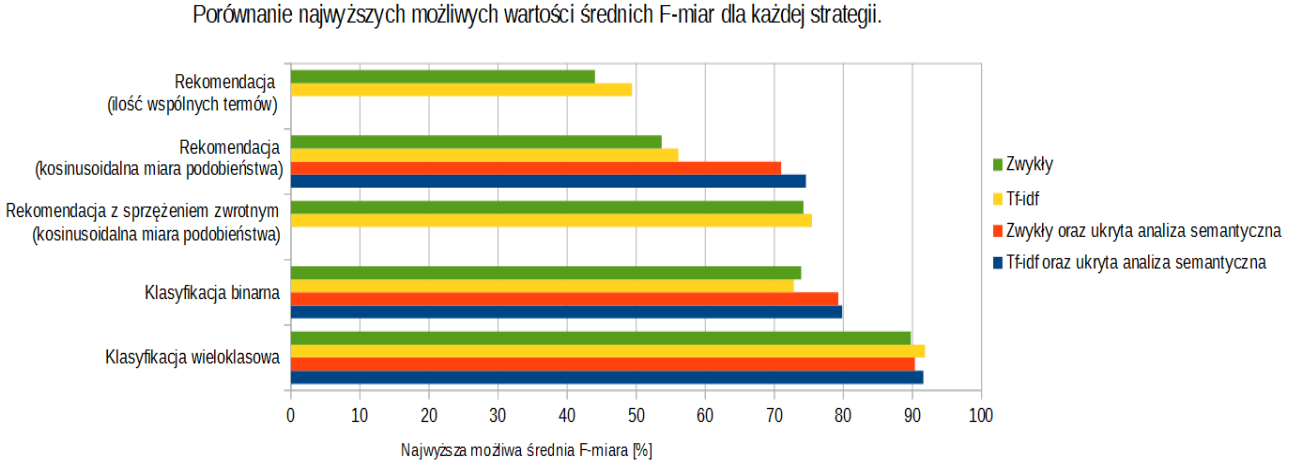
Badania również wykazały, iż stosowanie techniki Tf-idf w większości przypadków rzeczywiście poprawiło wyniki wyszukiwania, ponieważ zaletą techniki Tf-idf jest skupienie się na termach rzadko występujących w kolekcji, a zatem posiadających wysoką wartość dyskryminacyjną.

Badania dowiodły, że sprzężenie zwrotne znacząco poprawiło wyniki wyszukiwania dla rekomendacji z użyciem kosinusoidalnej miary podobieństwa. Warto również zauważyć, że rekomendacja z sprzężeniem zwrotnym osiągnęła porównywalne wyniki do klasyfikacji binarnej. Przyczyną tego był fakt, iż sprzężenie zwrotne jest rozwiązaniem bardzo zbliżonym do techniki tworzenia profilów metodą Rocchio. Użytkownik w trakcie udzielania feedbacku na temat relewancji zwracanych przez system dokumentów, generował wirtualny centroid, który posiadał podobne właściwości jak centroid użytkowy wybrany ręcznie przez użytkownika w przypadku klasyfikacji binarnej. Wyliczony centroid wirtualny umożliwił zatem łatwiejszy dostęp do informacji podobnych do centroidu.

Najgorszy wynik osiągnął algorytm do porównywania każdej pary dokumentów w oparciu o ilość wspólnych termów. Jego zaletą była jednak najmniejsza złożoność pamięciowa oraz obliczeniowa, ponieważ każdy dokument był reprezentowany tylko przez zbiór kilku najczęstszych termów, zamiast standardowego modelu w postaci macierzy incydencji term-dokument. Wadą tego rozwiązania było to, że niemożliwym było dokonanie redukcji macierzy term-dokument oraz skorzystanie z ukrytej analizy semantycznej dla powyższego algorytmu. Również niemożliwym było wyznaczenie rankingu dla zbioru dokumentów oznaczonych jako relewantne, ponieważ funkcja klasyfikująca dany dokument jako relewantny mogła zwrócić tylko dwie wartości – prawdę lub fałsz. Zatem niemożliwe było wyznaczenie szczegółowego rankingu dokumentów tak jak w przypadku rekomendacji z wykorzystaniem kosinusoidalnej miary podobieństwa.

Kontynuacją badań przeprowadzonych w niniejszej pracy dyplomowej, mogło by być przetestowanie kombinacji użycia ukrytej analizy semantycznej oraz sprzężenia zwrotnego. Badania wykazały, że każda z powyższych technik znacząco poprawiła wyniki wyszukiwania. Powyższa kombinacja nie została przetestowana z powodu zbyt licznej ilości parametrów konfiguracyjnych występujących łącznie dla obu technik.

Na rys 12.1 zaprezentowano końcowe porównanie każdej strategii z uzyskaną najlepszą możliwą dla niej konfiguracją, pozwalającą na uzyskanie najwyższej wartości średniej F-miary.



Rys. 12.1: Porównanie najwyższych możliwych wartości średnich F-miar dla każdej strategii. Opracowanie własne.

Bibliografia

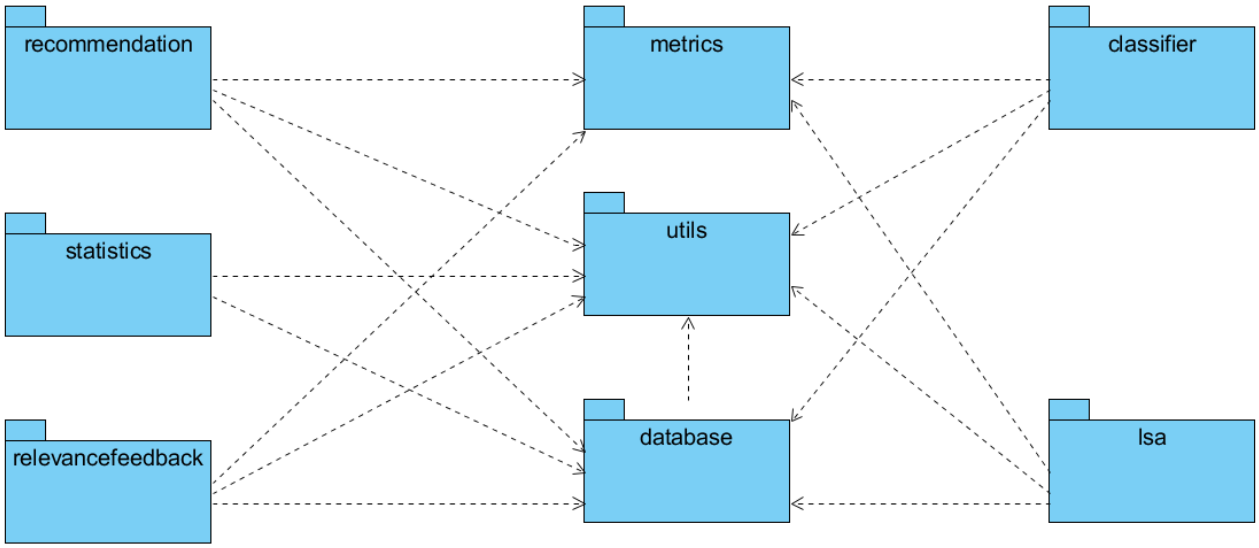
[1] – Stefanowicz B., *Informacja*, Warszawa , Oficyna Wydawnicza Szkoły Głównej Handlowej, 2010
[2] – Oleński J., *Standardy informacyjne w gospodarce*, Warszawa, Wydawnictwo Uniwersytetu Warszawskiego, 1997
[3] – Czekaja J., *Podstawy zarządzania informacją*, Kraków, Wydawnictwo Uniwersytetu Ekonomicznego w Krakowie, 2012
[4] – Etzel B., Thomas P., *Personal information management : tools and techniques for achieving professional effectiveness*, London, Macmillan Business, 1996
[5] – Perechuda K., *Zarządzanie wiedzą w przedsiębiorstwie*, Warszawa, Wydawnictwo Naukowe PWN, 2005
[6] - Mariusz Grabowski, Agnieszka Zając, *Dane, informacja, wiedza - próba definicji*, Kraków, Katedra Informatyki Akademii Ekonomicznej w Krakowie, 2013
[7] - Wang R., Strong D., *Beyond Accuracy: What Data Quality Means to Data Consumers*, Journal of Management Information Systems, Spring, 1996, tom 12, Nr. 4. s. 5-33.
[8] – Kowalski G., *Information Retrieval Architecture and Algorithms*, Springer, 2011
[9] - *Information storage and management : storing, managing, and protecting digital information*, EMC Education Services, 2009
[10] – Choroś K., *Metodologia Projektowania Systemów Informacyjnych, systemy informatyczne - wykłady*, Wrocław, Politechnika Wrocławska, 2012
[11] – Sammut C., Webb G., *Encyclopedia of Machine Learning*, Springer, 2010
[12] – Gliński W., *Wstęp do ontologii - wykłady*, Politechnika Warszawska, 2005
[13] – Trawiński B., *Zarządzanie projektem informatycznym, jakość danych - wykłady*, Politechnika Wrocławska, 2012
[14] – Nguyen N., Sobecki J., *Using consensus methods to construct adaptive interfaces in multimodal web-based systems*, Universal Access in the Information Society, Springer, 2003, Volume 2, Issue 4
[15] – Manning C., Raghavan P., Schütze H., *Introduction to Information Retrieval*, Cambridge University Press, 2008.
[16] – Bartkiewicz W., *Wprowadzenie do budowy usług informacyjnych - wykłady*, Uniwersytet Łódzki, 2010
[17] – *PostgreSQL documentation*, PostgreSQL Global Development Group, 2014 http://postgresql.org/docs/9.3/static/textsearch-dictionaries.html
[18] – Porter M., *The Porter Stemming Algorithm*, http://tartarus.org/martin/PorterStemmer/, 2006
[19] - Lee J., *Properties of Extended Boolean Models in Information Retrieval*, Daejeon, Korea Institute of Science and Technology, 1994
[20] – Shankland S., *We're all guinea pigs in Google's search experiment*, 2008 http://www.cnet.com/news/were-all-guinea-pigs-in-googles-search-experiment
[21] - Ide E., *Relevance Feedback in a Automatic Document Retrieval System*, Information Storage and Retrieval, Report No. ISR-15, Department of Computer Science, Cornell University, 1969
[22] – Yanai H., Takeuchi K., Takane Y., *Projection Matrices, Generalized Inverse Matrices, and Singular Value Decomposition*, Springer, 2011
[23] – Hedley J., *Java HTML Parser*, http://jsoup.org/, 2014
[24] – *Wikipedia contents categories*, Wikimedia Foundation, 2014 http://en.wikipedia.org/wiki/Portal:Contents/Categories,
[25] – Bush V., *As We May Think,* Atlantic Monthly, 1945

Spis tabel

Spis ilustracji

Załącznik 1 – Architektura zaimplementowanego systemu

Poniżej zaprezentowano podstawową architekturę zaimplementowanego systemu w ramach pracy dyplomowej.



Zaimplementowany system składał się z następujących modułów:

- recommendation - odpowiedzialny za przeprowadzenie badań oraz wygenerowanie raportów związanych z rekomendacją dokumentów tekstowych.
- statistics - odpowiedzialny za wygenerowanie raportów dotyczących danych statystycznych testowanej kolekcji dokumentów tekstowych.
- relevancefeedback - odpowiedzialny za przeprowadzenie badań oraz wygenerowanie raportów związanych z techniką sprzężenia zwrotnego.
- classifier - odpowiedzialny za przeprowadzenie badań oraz wygenerowanie raportów związanych z klasyfikacją binarną oraz wieloklasową dokumentów tekstowych.
- lsa - odpowiedzialny za przeprowadzenie badań oraz wygenerowanie raportów związanych z ukrytą analizą semantyczną.
- database – wykorzystywany przez wszystkie powyższe moduły, odpowiedzialny za obsługę połączenia z bazą danych.
- utils – wykorzystywany przez wszystkie powyższe moduły, odpowiedzialny za wykonanie podstawowych operacji takich jak wyznaczenie kosinusoidalnej miary podobieństwa oraz zapis wygenerowanych raportów do pliku.
- metrics – odpowiedzialny za wyliczenie wartości funkcji precyzji oraz zwrotu na podstawie podanych rekomendacji.

System posiadał także dodatkowy, startowy moduł o nazwie „main”, który uruchamiał tworzenie raportów dla powyższych modułów.

Załącznik 2 – Zawartość płyty

Płyta CD zawiera:

1. Folder o nazwie *praca* z elektroniczną wersją pracy w formacie pdf i odt.
2. Folder o nazwie *kod_zrodlowy* z kodem źródłowym zaimplementowanego systemu.
3. Folder o nazwie *baza_danych* zawierający pliki ze schematem bazy danych i danymi.
4. Folder o nazwie *wyniki*, zawierający arkusz kalkulacyjny w formacie xlsx.

Załącznik 3 – Schemat bazy danych

TERMY

Tabela przechowująca reprezentację wektorową danego dokumentu tekstowego oraz nazwę kategorii do której należy dokument. Do normalizacji dokumentów tekstowych użyto algorytmu stemmingu Snowball.

Pole	Typ	Opis
dokument	text	Nazwa dokumentu tekstowego
term	tsvector	Wektor termów wyekstrahowany z danego dokumentu
kategoria	varchar(15)	Nazwa kategorii do której należał dokument

Klucz unikalny: dokument

SQL

```
CREATE TABLE termy
(
  dokument text NOT NULL,
  term tsvector,
  kategoria character varying(15)
)
```

TERMY_ISPELL

Tabela przechowująca reprezentację wektorową danego dokumentu tekstowego oraz nazwę kategorii do której należy dokument. Do normalizacji dokumentów tekstowych użyto techniki lematyzacji z słownikiem morfologicznym Ispell.

Pole	Typ	Opis
dokument	text	Nazwa dokumentu tekstowego
term	tsvector	Wektor termów wyekstrahowany z danego dokumentu
kategoria	varchar(15)	Nazwa kategorii do której należał dokument

Klucz unikalny: dokument

SQL

```
CREATE TABLE termy_ispell
(
  dokument text NOT NULL,
  term tsvector,
  kategoria character varying(15)
)
```