Studia i Materiały Informatyki Stosowanej, Tom 4, Nr 6, 2012
str. 25-42

PRZEGLĄD METOD I TECHNIK EKSPLORACJI DANYCH TEKSTOWYCH

Marcin Mirończuk

Politechnika Białostocka
Wydział Elektryczny
ul. Wiejska 45A, 15-351 Białystok
e-mail: m.marcinmichal@gmail.com

Streszczenie : *W artykule opisano autorską klasyfikację metod i technik eksploracji danych tekstowych. Opisano aktualnie dostępne oraz stosowane metody reprezentacji danych tekstowych oraz techniki ich przetwarzania. Przeprowadzono także dyskusję na temat przetwarzania dokumentów za pomocą prezentowanych metod. Omówiono możliwości jak i ograniczenia poszczególnych prezentowanych metod do przetwarzania dokumentów tekstowych.*

eksploracja danych tekstowych, metody analizy danych tekstowych, eksploracyjna analiza danych tekstowych

Review of methods and text data mining techniques

Abstarct: *This article describes the author's classification of the methods and techniques of textual data mining. In this article also describes the currently available methods and sauces representation of textual data and their processing techniques. Also conducted a discussion on the processing of text documents using the presented methods. This paper also discussed the possibilities and limitations of individual methods to process the presented text documents.*

Keywords: *text data mining , methods of analysis of textual data, exploratory analysis of text data, text analyzing*

1. WSTĘP

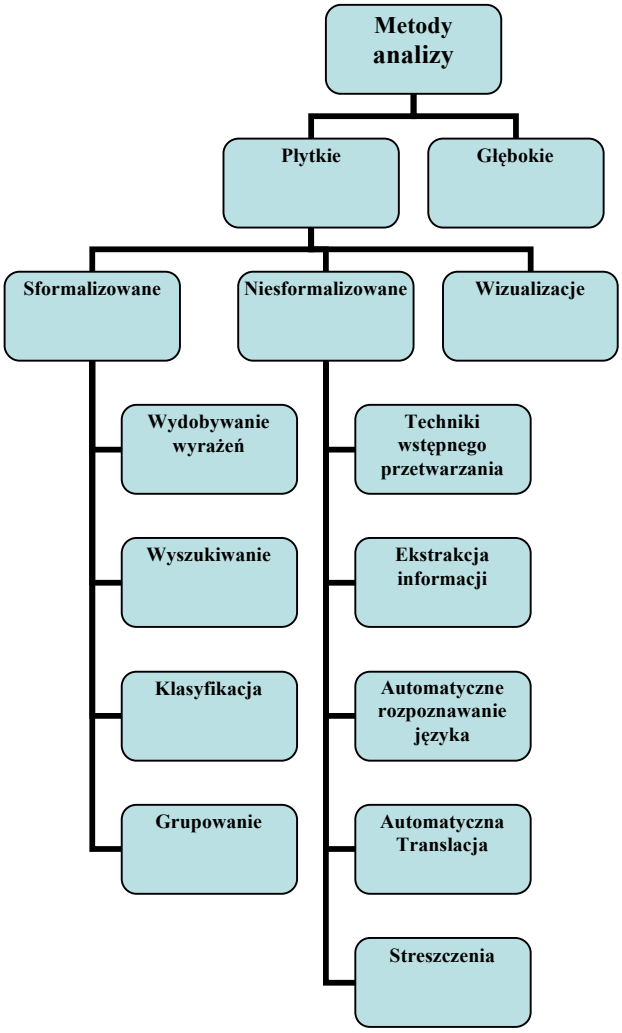
W badaniach dotyczących przetwarzania dokumentacji ze zdarzeń [1-4], pochodzącej z systemu ewidencji zdarzeń EWID-99 [4-8] przeznaczonego dla Państwowej Straży Pożarnej PSP, autor wykorzystuje metody oraz techniki z zakresu eksploracyjnej analizy danych tekstowych (*ang. text mining*). W publikacji [9] przedstawiono autorski przegląd i klasyfikację zastosowań, metod oraz technik z zakresu ogólnie pojętej eksploracji danych. W niemniejszej publikacji opisano szczegółowo wybraną gałąź tej klasyfikacji związaną z tekstowym źródłem danych [9] które stanowią dokumenty wyrażone za pomocą języka naturalnego.

Celem publikacji jest w szczególności przedstawienie czytelnikowi tzw. płytkich metod analizy tekstu. Aktualnie dostępna jest dość znaczna ilość publikacji i książek dotyczących głębokiego przetwarzania tekstów w języku polskim [10-13]. Natomiast ilość pozycji dotyczących płytkiej analizy tekstu jest znacznie ograniczona oraz nie omawia kompleksowo pod względem taksonomii tego zagadnienia [10, 14]. Z tych względów autor zaproponował i opisał własny podział metod z zakresu metod analizy tekstu a w szczególności metod służących do płytkiej analizy dokumentów tekstowych. Prezentacja rozważań nad metodami eksploracji danych tekstowych (punkt 2) została rozpoczęta od opisu aktualnie dostępnych i używanych reprezentacji dokumentów tekstowych (podpunkt 2.1). Następnie omówiono metody analizy dokumentów tekstowych niewymagające jak i wymagające (podpunkt 2.2) opisanych reprezentacji tekstu. W dalszej kolejności opisano metody wizualizacji wyników pochodzących
z przetwarzania tekstów (podpunkt (2.3). Na końcu przedstawiono podsumowanie oraz wnioski dotyczące proponowanej taksonomii oraz samej eksploracji danych tekstowych.

2. METODY EKSPLORACJI DANYCH TEKSTOWYCH

Dziedzina techniki zajmująca się przetwarzaniem komputerowym nieustrukturyzowanych danych w postaci

dokumentów tekstowych i wyciągania z nich informacji wysokiej jakości nazywa się eksploracją tekstu [15, 16]. W obrębie tej dziedziny powstało wiele nie do końca usystematyzowanych metod, technik oraz pojęć, które w niniejszym artykule zostały odpowiednio pogrupowane i szczegółowo omówione. Autorską taksonomię metod analizy tekstu przedstawia rysunek 1.



Rysunek 1 Taksonomia eksploracyjnych metod analizy tekstu. Źródło: [opracowanie własne]

W eksploracyjnej analizie tekstu dostępne są dwie metody przetwarzania tekstu: płytkie i głębokie. Pierwsza metoda dotycząca płytkiej analizy tekstu (*ang. shallow text processing – STP*), określa grupę działań polegających na rozpoznawaniu struktur tekstów nierekurencyjnych lub o ograniczonym poziomie rekurencji, które mogą być rozpoznane z dużym stopniem pewności. Struktury wymagające złożonej analizy wielu możliwych rozwiązań są pomijane lub analizowane częściowo. Analiza skierowana jest głównie na rozpoznawanie nazw własnych, wyrażeń rzeczownikowych, grup czasownikowych bez rozpoznawania ich wewnętrznej struktury i funkcji w zdaniu. Analiza dotyczy też głównie dużych zbiorów dokumentów tekstowych a nie pojedynczych dokumentów a także takich zagadnień jak m.in. klasyfikacja (kategoryzacja) dokumentów (*ang. document classification lub document categorization*) ich grupowania (*ang. dokument clustering*) i wyszukiwania z nich informacji (*ang. information retrieval – IR*) [17-19]. Celem tej analizy jest przyporządkowanie nieustrukturyzowanego tekstu wyrażonego za pomocą języka naturalnego do ustalonej reprezentacji (zazwyczaj składającej się ze zbioru obiektów). Przyporządkowanie to odbywa się na drodze procesu wykorzystującego specyficzne dla danej dziedziny algorytmy [19]. Druga metoda opiera się na tzw. głębokiej analizie tekstu (*ang. deep text processing – DTP*) i jest procesem komputerowej analizy lingwistycznej wszystkich możliwych interpretacji i relacji gramatycznych występujących w tekście naturalnym. Zazwyczaj jest bardzo złożona i z reguły dotyczy pojedynczego dokumentu. Pomija się wszelkie zależności statystyczne i stosuje się rozwiązania polegające na przetwarzaniu danych w oparciu o predefiniowane wzorce lub gramatyki [10, 19].

2.1. Reprezentacja dokumentów tekstowych

Aktualnie rozwinięte i wykorzystywane praktycznie są dwie reprezentacje dokumentów tekstowych: reprezentacja wektorowa oraz reprezentacja grafowa. Obie z nich zostały omówione w podpunktach 2.1.1 oraz 2.1.2.

2.1.1. Model wektorowy reprezentacji dokumentów tekstowych

Model wektorowy reprezentacji dokumentów tekstowych polega na przedstawieniu ich w postaci przestrzenno-wektorowego opisu (modelu wektorowego, *ang. vector space model – VSM*). Dokumenty i występujące w nich wyrażenia, są reprezentowane w postaci macierzy. Powszechnie, za wyrażenie w reprezentacji przestrzenno-wektorowej, uważane jest jedno wyrażenie np. *pożar* lub para wyrażeń np. *mocne zadymienie*. Zazwyczaj nie są to wszystkie możliwe wyrażenia – zwykle w etapie wstępnego przetwarzania

(*ang. preprocessing*) dokonuje się ich selekcji (za pomocą metod opisanych w podpunkcie 2.2.1 i 2.2.2) oraz oceny ich istotności dla modelowanej dziedziny.

Rysunek 2 przedstawia macierzową postać zapisu dokumentów i związanych z nimi wyrażeń. Dokumenty reprezentowane są poprzez wiersze (*m*), natomiast wyrażenia znajdują się w kolumnach (*n*) macierzy *A* zwanej macierzą dokumentów-wyrażeń (*ang. term-document matrix*). Bardziej ogólnym pojęciem stosowanym w lingwistyce komputerowej jest *korpus* określający dużą kolekcję dokumentów, opisanych i sprowadzonych np. w szczególnym przypadku do opisywanej postaci macierzowej (rysunek 2). W niniejszym tekście *korpus* będzie równoważny macierzy *A*.

$$A = \begin{bmatrix} w_{11} & . & . & . & w_{1j} \\ . & . & & & . \\ . & & . & & . \\ . & & & . & . \\ w_{i1} & . & . & . & w_{ij} \end{bmatrix}, A \in R^{m \times n}$$

$$Gdzie:$$

$$1 \le i \le m$$

$$1 \le j \le n$$

Rysunek 2 Struktura reprezentacji przestrzenno-wektorowej dokumentów . Źródło: [opracowanie własne na podstawie [20]]

W rozwiązaniach praktycznych ilość wierszy macierzy *A* jest znacznie większa od ilości wyrażeń (m >> n). Do poprawy przetwarzania, wydajniejszego składowania takiej struktury w systemach informatycznych i analizy stosuje się konwencję odwróconą tj. w wierszach zapisywane są wyrażenia natomiast w kolumnach dokumenty. Wówczas taki zapis nosi nazwę *pliku odwróconego* a jego sposób indeksowania wyrażony jest poprzez *indeks odwrotny* [21, 22]. Element macierzy w_{ij} oznacza wagę, a tym samym znaczenie *j*-tego wyrażenia w *i*-tym dokumencie (rysunek 2 reprezentuje taki zapis). W zależności od sposobu kodowania informacji zawartej w elemencie w_{ij} czyli w wadze wyrażenia lub bardziej precyzyjnie w wartościach składowych wektora wyrażeń, istnieje możliwość otrzymania różnych odmian reprezentacji przestrzenno-wektorowej tekstu. Do popularnych, stosowanych w praktyce odmian zaliczamy m.in. reprezentacje boolowską (binarną), częstotliwościową występowania wyrażeń (*ang. term frequency – TF*), odwrotną częstość dokumentu (*ang. inverse-document-frequency – IDF*), mieszaną TF-IDF, logarytmiczną, ważoną logarytmiczną, okapi BM25 oraz probabilistyczną [10, 20, 23-25]. Reprezentacja:

a) boolowska (binarna) – występuje wówczas, kiedy zostanie odnotowany fakt zaistnienia *j*-tego wyrażenia w *i*-tym dokumencie, natomiast nie precyzuje ona liczby wystąpień. Element w_{ij} macierzy *A* przyjmuje wartość 1 (*j*-te wyrażenie znajduje się w *i*-tym dokumencie) lub 0 (*j*-te wyrażenie nie znajduje się w *i*-tym dokumencie),

b) częstotliwościowa występowania wyrażeń (*ang. term frequency – TF*) – występuje wówczas, kiedy oprócz odnotowania faktu zaistnienia *j*-tego wyrażenia w *i*-tym dokumencie zostanie określona także jego częstość, czyli liczba jego wystąpień w zadanym dokumencie,

c) odwrotnej częstości dokumentu (*ang. inverse-document-frequency – IDF*) – polegającą na tym, iż poszczególne wagi w_{ij} wyrażone są za pomocą wyrażenia $\log(N/n_j)$, gdzie: *N* reprezentuje liczbę wszystkich dokumentów zaś n_j liczbę dokumentów z *j*-tym wyrażeniem,

d) mieszana TF-IDF – występuje wówczas, gdy pomnożone zostaną przez siebie wagi w_{ij} wyrażone za pomocą ww. schematu TF i IDF, czyli mieszana reprezentacja TF-IDF równa jest $TF \cdot IDF$,

e) logarytmiczna – występuje wówczas, gdy następuje zastąpienie wszystkich niezerowych elementów macierzy *A* wartościami w_{ij} równymi $1+\log(w_{ij})$,

f) ważona logarytmiczna – występuje wówczas, gdy następuje zastąpienie wszystkich niezerowych elementów macierzy *A* wartościami w_{ij} obliczonymi za pomocą następującej formuły $(1+\log(w_{ij})) \cdot \log(\frac{N}{n_j})$,

g) okapi BM25 – stosowana jest w przypadkach długich dokumentów tekstowych, gdzie prawdopodobieństwo, że dany wyraz pojawi się wiele razy jest wysokie. Powoduje to wzrost wartości wagi *TF* co w efekcie sprawia, że długie dokumenty są bardziej „faworyzowane". BM25 to rodzina funkcji wykorzystywana do obliczenia wagi w_{ij} z uwzględnieniem długości dokumentów. Mając dokument *d* (od angielskiego słowa *document*) i wyrażenie *t* (od angielskiego słowa *term*) można obliczyć wagę korzystając z zależności:

$$bm25(d,t) = idf \cdot \frac{f(t,d) \cdot (k_1 + 1)}{f(t,d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{avg(d)}} \quad (1)$$

Gdzie:
- f(t,d) – liczba wystąpień wyrażenia *t* w dokumencie *d*
- |d| – długość dokumentu *d*
- avg(d) – średnia długość dokumentu w kolekcji
- k_1 i b – wartości stałe (przeważnie przyjmuje się $k_1 = 1.2$ i $b = 0.75$)
- idf – zmodyfikowany schemat IDF wyrażony w postaci formuły $idf(t) = \log\frac{N - n(t) + 0.5}{n(t) + 0.5}$, *N* oznacza liczbę wszystkich dokumentów, a *n(t)* liczbę dokumentów zawierających wyrażenie *t*.

h) probabilistyczna – występuje wówczas, gdy waga w_{ij} wyrażenia *t* w dokumencie *d* zostanie oszacowana na podstawie zdarzenia losowego, polegającego na wystąpieniu danego wyrażenia *t* w dokumencie *d* pod warunkiem modelu *M*. Model *M* zawiera informacje na temat korpusu *A* tj. całkowitą ilość wyrażeń oraz częstotliwość występowania poszczególnych wyrażeń w korpusie *A*. Proste oszacowanie prawdopodobieństwa wystąpienia wyrażenia *t* można dokonać zgodnie z zasadą estymacji największej wiarygodności (*ang. maximum likelihood estimation – MLE*) [23]:

$$w_{ij} = P_{ML}(Y_i = t \mid d, M) = \frac{TF_{t,d}}{\sum_{t \in d} TF_{t,d}} \quad (2)$$

Wzór na wagę wyrażenia przedstawiony w postaci estymacji największej wiarygodności można interpretować następująco:

$$\mathrm{P_{ML}}(Y_i = t \mid d, M) = \frac{\begin{array}{l}\text{czestotliwosc}\\ \text{wystepowania}\\ \text{wyrazenia t}\\ \text{w dokumencie d}\end{array}}{\begin{array}{l}\text{suma wszystkich}\\ \text{czestotliwosci}\\ \text{wyrazen t}\\ \text{w dokumencie d}\end{array}} \quad (3)$$

Na podstawie macierzy *A* z odpowiednio skonstruowanymi wagami w_{ij} możliwe jest więc wyznaczenie podobieństwa słów oraz dokumentów. Podobieństwo słów wyrażane jest poprzez określenie podobieństwa odpowiadających im kolumn tej macierzy, natomiast o podobieństwie dokumentów wnioskuje się na podstawie analizy podobieństwa wierszy tej macierzy. Najczęściej wszystkie wagi w_{ij} wektorów macierzy *A* w zastosowaniach praktycznych są normalizowane do 1.

Wprowadzenie wektorowo-przestrzennego modelu dokumentów umożliwia matematyczną analizę zagadnienia np. wyszukiwania dokumentów tekstowych. Zagadnienie wyszukiwania zostało omówione w podpunkcie 2.2.2. Budowanie reprezentacji tekstu na samych wyrażeniach jest jednak często mocno ograniczone. Do zasadniczych wad tego modelu należą:
- utrata wszelkiej informacji na temat struktury dokumentów: tytuł, nagłówki etc.,
- pominięcie informacji na temat kolejności słów a więc i związków między nimi (występowanie wyrażeń jest niezależne od siebie),
- istnieje konieczność wyboru wyrażeń, dla których zostanie stworzona macierz – liczba wymiarów musi być z góry znana.

Ze względu na ww. ograniczenia, proponowane są takie rozwiązania aby składowymi wektora reprezentującego dokument były automatycznie wydobyte cechy tekstu (jak język, styl, itp.) zamiast wyrażeń kluczowych oraz elementy wydobyte z semantycznego zbioru, a więc wyrażenia i powiązania między nimi skoncentrowane na stronie znaczeniowej tekstu [17]. Ograniczenia związane z reprezentacją przestrzenno-wektorową wymogły stosowanie drugiego sposobu reprezentacji dokumentów tekstowych, a mianowicie ich opis grafowy. Należy zaznaczyć, że reprezentacja przestrzenno-wektorowa mimo ograniczeń posiada też zalety. Sprawiają one, że jest ona dalej powszechnie stosowaną i badaną reprezentacją dokumentów tekstowych. Zaletą wyboru takiej reprezentacji dokumentów jest jej zbieżność z reprezentacją stosowaną typowo w uczeniu maszynowym (obiekty opisane za pomocą atrybutów), dzięki czemu można do niej zastosować istniejące metody z tej dziedziny. Badania także dowodzą, że niektóre relacje semantyczne mogą zostać wydobyte z tekstu z dużą dokładnością z pominięciem kolejności słów [26], natomiast związek pomiędzy wyrażeniami może zostać ustalony za pomocą analizy współwystępowania wyrażeń [27]. Również wykonywanie operacji, takich jak: liczenie odległości, przeprowadzanych na wektorach, jest aplikacyjnie łatwiejsze w realizacji i bardziej efektywne obliczeniowo od konkurencyjnej reprezentacji, np. opartej na modelu grafowym.

2.1.2. Model grafowy reprezentacji dokumentów tekstowych

Model grafowy reprezentacji tekstu bazuje na teorii grafów. Model ten nazywany jest także modelem ustrukturyzowanym. Początkowo został on zaproponowany do analizy stron ogólnoświatowej sieci (*ang. World Wide Web – WWW*) a potem do opisu, analizy i poddawania procesom np. klasyfikacji [28] dokumentów z pewną strukturą [29]. Podstawowym założeniem wprowadzenia reprezentacji grafowej była chęć przeciwdziałania mankamentom związanym z reprezentacją przestrzenno-wektorową. Dzięki zastosowaniu grafów do opisu dokumentów tekstowych możliwe stało się przechowywanie informacji m.in. o: związkach wynikających z kolejności wyrazów, charakterystykach opisywanych w dokumentach obiektów, relacjach między nimi oraz zależnościach przyczynowo-skutkowych.

Schenker i współautorzy [30] zaproponowali podejście modelowania całego dokumentu tekstowego jako grafu połączeń między wyrażeniami. W swojej pracy przedstawili następujące sposoby reprezentacji dokumentu: standardowa (*ang. standard representation*), prosta (*ang. simple representation*), *n*-odległości (ang. *n-distance representation*), prostej *n*-odległości (ang. *n-simple distance*), bezwzględnej częstości (ang. *absolute frequency*), względnej częstości (ang. *relative frequency*) [30, 31]. Kolejno poszczególne sposoby reprezentacji dokumentów definiowane są następująco:

a) reprezentacja standardowa (*ang. standard representation*) – dla każdego wyrażenia tworzony jest węzeł, przy czym jedno wyrażenie występuje tylko raz w grafie dla dokumentu. Dokument jest podzielony na sekcje: tytuł (wraz z metadanymi), odnośniki (tekst w odnośnikach), tekst (cały widoczny tekst, włącznie z odnośnikami). Jeżeli dwa wyrażenia występują bezpośrednio po sobie w obrębie jednej sekcji to jest tworzony łuk skierowany od pierwszego do drugiego z nich. Łuk jest oznaczony zgodnie z miejscem występowania jako tytuł (*ang. title – TI*), powiązanie (*ang. links – L*) lub tekst (*ang. text – TX*). Po zbudowaniu grafu wyrażenia sprowadzane są do rdzeni morfologicznych (*ang. stemming)* a węzły są zwijane do najczęściej występującej formy,

b) reprezentacja prosta (*ang. simple representation*) – analogiczna do reprezentacji standardowej z tą różnicą, że przetwarzany jest tylko tekst widoczny na stronie a do łuków nie są przypisywane etykiety,

c) reprezentacja *n*-odległości (ang. *n-distance representation*) – łuki grafu są tworzone nie tylko dla wyrażeń występujących bezpośrednio po sobie, ale również dla *n* wyrażeń do przodu (*n* jest parametrem dostarczanym przez użytkownika). Połączenie między wyrażeniami jest tworzone tylko wtedy, gdy nie zostaną napotkane predefiniowane znaki interpunkcyjne. Łuk jest etykietowany odległością pomiędzy słowami,

d) reprezentacja prostej *n*-odległości (ang. *n-simple distance*) – analogiczna reprezentacja do *n*-odległości, z tą różnicą, że łuki nie są etykietowane odległością. Graf mówi tylko o tym, że pomiędzy wyrażeniami występuje połączenie, ale nie mówi jak jest ono silne,

e) reprezentacja bezwzględnej częstości (ang. *absolute frequency*) – podobna do reprezentacji prostej – węzły są tworzone dla wyrażeń występujących bezpośrednio po sobie, nie są uwzględniane informacje strukturalne. Do węzła przypisywana jest ilość wystąpień wyrażenia w dokumencie, do łuku – częstość wystąpienia dwóch wyrażeń po sobie,

f) reprezentacja względnej częstości (ang. *relative frequency*) – analogicznie do reprezentacji bezwzględnej częstości, przy czym ilość wystąpień wyrażenia (etykiety węzłów) są normalizowane przez maksimum z częstości wszystkich węzłów, a ilość powiązań miedzy wyrażeniami (etykiety łuków) przez maksimum liczebności wszystkich powiązań.

Model grafowy często powiązany jest z *ontologią*, rozumianą jako formalny sposób opisu wyodrębnionego fragmentu rzeczywistości [32]. Definicja ontologii obejmuje opis obiektów występujących w rzeczywistości oraz opis zależności pomiędzy nimi. Pod tym względem możliwe więc jest aby reprezentatywne, wybrane wyrażenia z grafu stały się obiektami z ontologii lub klasami z *hierarchii klas obiektów*. Wśród zależności występujących pomiędzy reprezentowanymi w ontologii obiektami szczególnie ważną role odgrywają relacje semantyczne np. *zawiera, obejmuje, posiada*. Z tego powodu układ obiektów wraz z opisem występujących pomiędzy nimi relacji semantycznych nazywa się siecią semantyczną. Dogodną strukturą do reprezentowania takich sieci są grafy, kraty (*ang. lattice*) jak i hierarchie klas obiektów. Wyrażenie *hierarchie klas obiektów* należy traktować jako termin z dziedziny programowania obiektowego [33].

Wadą reprezentacji grafowej jest znacznie mniejszy wachlarz metod analitycznych przystosowanych do operowania na informacjach przechowywanych przy wykorzystaniu złożonych struktur danych [34]. Ograniczenia te w szczególności związane

z przechowywaniem danych, powoli przestają mieć znaczenie ze względu na opracowywany prototypowy model zorientowany koncepcyjnie, przystosowany do przechowywania struktur zagnieżdżonych [35-37]. Model zorientowany koncepcyjnie lub model zorientowany na pojęcia (*ang. concept oriented model – COM*) zaproponowany został przez Savinova w 2004 [36]. Model ten stanowi nowe podejście do modelowania danych i bazuje na trzech głównych zasadach [37, 38]: zasadzie dwoistości (*ang. duality principle*) mówiącej, że każdemu elementowi (pojęciu) przypisana jest tożsamość (*ang. identity*) oraz encja (*ang. entity*), zasadzie włączenia (*ang. inclusion principle*) dotyczącej używania hierarchicznej struktury dla modelowania tożsamości oraz zasadzie porządku (*ang. order principle*) która mówi o używaniu matematycznej zasady porządku częściowego (*ang. partial order*) do reprezentowania semantyki danych. W przypadku modelu grafowego *konceptami* z modelu COM mogą być wybrane wyrażenia z modelu grafowego.

2.2. Metody analizy tekstu

Metody płytkiej analizy tekstu można podzielić ze względu na to czy do ich działania potrzebna jest sformalizowana reprezentacja dokumentu opisana w podpunktach 2.1.1 i 2.1.2 czy też nie. Przykład sformalizowanej reprezentacji tekstu stanowi reprezentacja wektorowa opisana w podpunktach 2.1.1 i 2.1.2. Niesformalizowana reprezentacja natomiast nie wymaga żadnej z powyższych reprezentacji. Metodami, które nie wymagają sformalizowanej reprezentacji, są: wstępne przetwarzanie tekstu, ekstrakcja informacji, automatyczne rozpoznawanie języka, automatyczna translacja tekstów. Metody te kolejno zostały omówione w podpunkcie 2.2.1. W przypadku sformalizowanych reprezentacji tekstu do metod jego analizy zaliczane są: wydobywanie wyrażeń z tekstów, wyszukiwanie informacji w szczególności wyszukiwanie informacji w reprezentacji przestrzenno wektorowej oraz grafowej, klasyfikacja oraz grupowanie. Metody te zostały omówione w podpunkcie 2.2.2.

2.2.1. Metody analizy bezpośredniej na tekście

Metodami, które nie wymagają sformalizowanej reprezentacji tekstu, są: wstępne przetwarzanie tekstu (podpunkt 2.2.1.1), ekstrakcja informacji (podpunkt 2.2.1.2), automatyczne rozpoznawanie języka (podpunkt 2.2.1.3), automatyczna translacja tekstów (podpunkt 2.2.1.4) oraz streszczenia dokumentów tekstowych (podpunkt 2.2.1.5).

2.2.1.1. Techniki wstępnego przetwarzania dokumentów tekstowych

Do technik wstępnego przetwarzania dokumentów tekstowych należą: ekstrakcja rdzeni wyrażeń (*ang. stemming*), tagowanie (*ang. tagging*), lematyzacja, usuwanie słów ze stop listy, przycinanie (*ang. pruning*) [10, 18]. Operacje te podejmowane są zanim dokument lub grupa dokumentów tekstowych zostanie przesłana do głównego procesu analizy np. wyszukiwania pełnotekstowego (*ang. full text serach*) [39] czy też innych metod przetwarzania tekstu. Przedstawione terminy, związane ze wstępnym przetwarzaniem tekstu, można zdefiniować w następujący sposób [10, 18]:

a) ekstrakcja rdzeni wyrażeń (*ang. stemming*) – określa znajdowanie tematów słów lub tych ich fragmentów, które są niezmienne dla wszystkich form,

b) tagowanie (*ang. tagging*) – oznacza wybór opisu morfo-składniowego, który jest właściwy w konkretnym kontekście użycia danej formy,

c) lematyzacja – jest to analiza morfologiczna ograniczana do znalezienia podstawowej formy wyrazu (identyfikacja leksemu),

d) usuwanie słów ze stop listy – na stop liście umieszcza się wyrażenia, które występują zbyt często by ich użycie jako kluczy wyszukiwania było celowe. Wyrażenia umieszczone na stop liście słów są odrzucane (filtrowane) podczas wczytywania dokumentu,

e) przycinanie (*ang. pruning*) – polega na usuwaniu niepotrzebnych słów, operacja ta ma na celu polepszenie skuteczności klasyfikacji. Można usuwać wyrażenia występujące najczęściej (*ang. most frequent*) i najrzadziej (*ang. least frequent*).

Wszystkie wyżej wymienione zabiegi stosuje się w celu ulepszenia przeprowadzanej analizy dokumentów tekstowych oraz ich wydajniejszego indeksowania. Zabiegi te stosowane w kontekście analizy tekstu pozwalają na identyfikację początkowego zestawu cech, który może być później ograniczony (i zoptymalizowany) w procesie wydobywania wyrażeń (podpunkt 2.2.2).

2.2.1.2. Ekstrakcja informacji

Ekstrakcja informacji (*ang. information extraction – IE*) jest to identyfikacja, polegająca na odnajdywaniu właściwej informacji w nieustrukturyzowanych danych tekstowych wyrażonych za pomocą języka naturalnego. Proces ten jest zgodny z klasyfikacją polegającą na strukturyzowaniu poprzez nadawanie klas semantycznych dla wybranych elementów tekstu. Proces ten czyni informację zawartą w

tekście bardziej właściwą i przydatną w realizowanych zdaniach [40]. Ekstrakcja informacji nazywana jest także ekstrakcją (rozpoznawaniem) encji i modelowania ich relacji (*ang. concept/entity extraction, named entity recognition*) [41], jednak jest to ograniczenie definicji ekstrakcji informacji tylko do jednego z podstawowych jej zadań. Wymienione zadanie polega na pozyskiwaniu z dokumentów tekstowych nazw obiektów np. osób oraz na wyznaczaniu związków i relacji pomiędzy wydobytymi obiektami. W ogólnym przypadku można pozyskiwać w ten sposób z tekstu nazwy miast, imiona i nazwiska osób, kody pocztowe, numery PESEL itp. W przypadku szczególnym, który stanowią analizy raportów z akcji ratowniczo-gaśniczych, można pozyskać informacje na temat: ilości akcji, w których brała udział dana osoba, ilości ofiar śmiertelnych zarejestrowanych w akcji ratunkowej. Przy pomocy tak wydobytych cech można sprawdzać czy analizowany obiekt np. osoba nie zmieniła rangi (nie awansowała na wyższy stopień), czy nie zaszły jakieś kluczowe zmiany na obiekcie np. niedziałające hydranty, czy też w przestrzeni mediów nie pojawiły się informacje o zdarzeniach określonego typu (katastrofy, wypadki, akty terrorystyczne). Do pozostałych podstawowych zadań z zakresu ekstrakcji informacji należą: rozróżnianie wyrażeń rzeczownikowych z relacją gramatyczną (*ang. noun phrase coreference resolution*), rozpoznawanie ról semantycznych (*ang. semantic role recognition*), rozpoznawanie relacji między encjami (*ang. entity relation recognition*) czy też rozpoznanie czasu oraz określanie linii czasu zachodzenia zdarzeń (*ang. timex and time line recognition*) [40].

Do typowych problemów, które muszą być rozwiązane przez system ekstrakcji informacji, należą następujące zagadnienia [10, 40]:

a) rozpoznanie i utworzenie skryptów (scenariuszy) będących kompleksowym opisem zdarzeń,
b) utworzenie modeli (wzorców) wynikających z tekstu,
c) podział tekstu na ciągi zdań,
d) podział zdań na wyrażenia z przypisanymi wartościami cech gramatycznych,
e) rozpoznawanie skrótów, fraz rzeczownikowych, nazw bez wnikania w ich strukturę wewnętrzną i ich funkcje w zdaniu,
f) budowanie przybliżonej struktury zdania (np. drzewa rozbioru) ze słów i wcześniej rozpoznanych elementów,
g) wypełnienie przygotowanych modeli informacjami z tekstu.

Pierwsze cztery ww. zadania mają charakter ogólny i ich rozwiązania mogą być stosowane w wielu różnych systemach. Ostatnie zadanie natomiast jest ściśle związane z konkretnym zastosowaniem. Wzorce i reguły ich wypełniania zależą od tego, jakich informacji poszukujemy.

Przytoczone wyżej pojęcia ekstrakcji informacji wiążą się najczęściej z normalizacją i identyfikacją w tekście wybranych typów danych oraz ich powiązań. Niemniej w skład tej metody można zaliczyć podejścia i zabiegi stosowane do wydobywania wyrażeń (cech) reprezentatywnych, od jakości których zależą np. wyniki wyszukiwania informacji dla dokumentu czy też ich grupy. W kontekście analizy tekstu i niniejszego opracowania cecha (*ang. feature*) znaczeniowo traktowana jest jako wyrażenie (*ang. term*). W dalszej kolejności, oprócz samego wydobywania, wyrażeń można też ekstrahować semantykę tych wyrażeń za pomocą np. analizy opartej o dane z korpusu lingwistycznego (reprezentacji przestrzenno-wektorowej dokumentów) [31]. Ogólnie do obu tych celów mogą służyć metody grupujące opisane w podpunkcie 2.2.2, jeżeli zadanie grupowania zostanie zdefiniowane na mniejszym poziomie ziarnistości niż dokument, a mianowicie na poziomie wyrażeń.

2.2.1.3. Automatyczne rozpoznawanie języka

Automatyczne rozpoznawanie języka (*ang. automatic language identification – ALI*) polega na identyfikacji wersji językowej dokumentu, w szczególności dokumentu tekstowego, który może zostać napisany w więcej niż jednym języku [42]. Do automatycznej identyfikacji wersji językowej wykorzystywane są głównie dwa rodzaje rozwiązań. Pierwsze rozwiązanie bazuje na statystycznym modelu języka i polega na oszacowaniu prawdopodobieństwa (*ang. estimate the probability*), że dana wejściowa próbka tekstu jest napisana w zadanym języku. Drugie rozwiązanie polega na porównaniu pomiędzy częstotliwością używanych wspólnych słów lub wyrażeń w próbce tekstowej z częstotliwością wydobytą ze statystycznej analizy dużego korpusu służącego jako odniesienie.

Automatyczne rozpoznawanie języka wykorzystywane jest najczęściej w sieci internetowej do analizowania wersji językowych stron internetowych, czy też korespondencji email. Pewne jego elementy mogą też być wykorzystane we wstępnym procesie tekstowej eksploracji danych w celu polepszenia jakości analizy.

2.2.1.4. Automatyczna translacja tekstów

Automatyczna translacja tekstów nazywana także tłumaczeniem maszynowym TM, polega na dokonywaniu

przekładu z jednego języka na drugi. Pierwsze próby TM były podejmowane w latach 50-tych. W latach 70-tych dziedzina ta przeżyła swój rozkwit ze względu na gwałtowny rozwój sprzętu jak i oprogramowania komputerowego. Do automatycznego tłumaczenia tekstu podchodzi się dwojako tj. dokonuje się tłumaczenia „zgrubnego", przeznaczonego do poprawiania przez człowieka (mamy tutaj do czynienie raczej ze wspomaganiem tłumaczenia, a nie z samym tłumaczeniem) oraz tłumaczenia ograniczonego do wąskiego podzbioru języka (np. prognozy pogody, raportów giełdowych) [10].

Największym problemem w tłumaczeniu i kluczem do jego sukcesu jest prawidłowe tłumaczenie słów a raczej ich znaczeń. Mimo pojawiających się problemów związanych z TM, w dalszym ciągu budzi ono wielkie zainteresowanie zarówno w środowisku naukowców jak i biznesowym [43-45].

2.2.1.5. Streszczenia dokumentów tekstowych

Streszczenia (podsumowania) dokumentów tekstowych (*ang. text document summarization*) polegają na wytwarzaniu streszczenia z obszernego dokumentu lub ich grupy [22, 34, 46]. Przykładowy algorytm bada powiązania między wyrażeniami. Jeżeli następuje odwołanie kilku wyrażeń do danego wyrażenia, wówczas zachodzi zwiększenie jego pozycji w rankingu. Jako podsumowanie analizy wyświetlane jest *n* zdań o najwyższym rankingu, tworząc w ten sposób streszczenie. Zagadnienie streszczenia dokumentów może zostać sprowadzone do podejścia selekcje cech ze względu na zastosowane techniki uczenia: uczenie nadzorowane lub nienadzorowane. Uczenie nadzorowane polega na ekstrakcji cech z odpowiednio dużego oznaczonego korpusu tekstowego (mamy dostęp do predefiniowanych klas cech) [47, 48]. Uczenie nienadzorowane natomiast polega na uchwyceniu pewnych właściwości tekstu, które umożliwią wydobycie wyrażeń kluczowych dla danego dokumentu lub ich grupy. W przypadku zastosowania uczenia nienadzorowanego możliwe jest podejście lokalne lub globalne. Przypadek lokalny, w kontekście analizy tekstu, występuje wówczas, gdy w procesie wydobywania słów kluczowych wykorzystywana jest tylko informacja o dostępnej grupie dokumentów lub pojedynczym dokumencie. Przykładowy algorytm wydobywania słów kluczowych oparty o tylko jeden dokument tekstowy, bez wykorzystania całego korpusu tekstów, zaproponowali Matsno i Ishizuka [27]. Podejście globalne bazuje natomiast, przy wydobywaniu wyrażeń kluczowych, na informacji o grupie dokumentów jak i całego korpusu. Propozycja analizy tekstu opartej o metodę globalną została opisana w pracy [49].

2.2.2. Metody analizy sformalizowanych reprezentacji tekstu

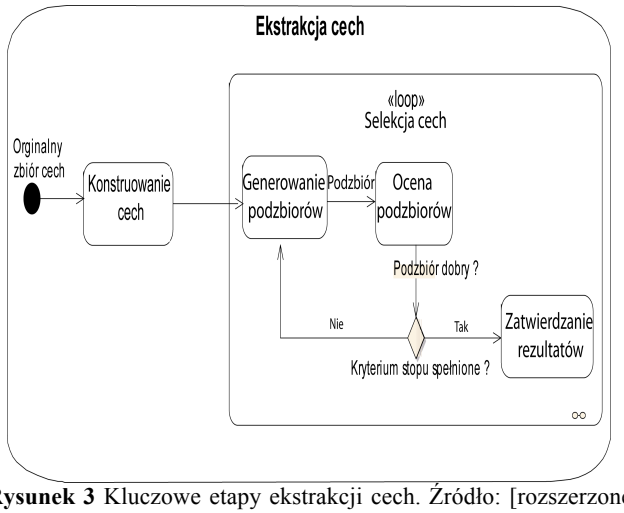
W przypadku sformalizowanych reprezentacji tekstu do metod jego analizy zaliczane są: wydobywanie wyrażeń z tekstów (podpunkt 2.2.2.1), wyszukiwanie informacji w szczególności wyszukiwanie informacji w reprezentacji przestrzenno wektorowej oraz grafowej (podpunkt 2.2.2.2), klasyfikacja (podpunkt 2.2.2.3) oraz grupowanie (podpunkt 2.2.2.4).

2.2.2.1. Wydobywanie wyrażeń

Wydobywanie wyrażeń może następować poprzez ich ekstrakcje (*ang. feature extraction*). Ekstrakcja cech w literaturze określana jest także jako transformacja cech (*ang. feature transform*) czy też generowanie, uogólnianie cech (*ang. feature generation*). Proces ekstrakcji cech podzielony jest na dwa etapy: konstruowania cech (*ang. feature construction*) a następnie ich selekcji (*ang. feature selection*) [50, 51]. Selekcja cech w literaturze określana jest także jako: selekcja zmiennych (*ang. variable selection*), redukcja cech (*ang. feature reduction*), selekcja atrybutów (*ang. attribute selection*), lub selekcja podzbioru zmiennych (*ang. variable subset selection*). Metody selekcji cech można rozpatrywać w kontekście dziedziny nauki związanej z uczeniem maszynowym (*ang. machine learning*), wówczas otrzymany zostanie dodatkowy podział (ze względu na zastosowane kryterium oceny podzbioru cech), na który składają się podkategorie: filtry (*ang. filters*), opakowywacze (*ang. wrapper*) i metody wbudowane (*ang. embedded methods*).

Metody ekstrakcji, nie tylko samych wyrażeń lecz i ich semantyki, są oparte na *hipotezie dystrybucyjnej* [52] i stanowią specyficzną odmianę metod ekstrakcji specjalnie stworzonych na potrzeby analizy tekstów. Metody wydobywania podobieństwa semantycznego wyrażeń z tekstów opierają się na uzyskaniu funkcji podobieństwa semantycznego. Przegląd takich metod, odwołania do nich i opisy można znaleźć w pracy [31].

Powyżej zostały opisane klasyfikacyjne „statyczne" aspekty ekstrakcji cech. Na proces ekstrakcji cech można spojrzeć w sposób dynamiczny, wyrażony w postaci algorytmu i automatu z określoną ilością sekwencji (stanów, etapów), którego działanie ma przynieść wydobycie interesujących składowych. Kluczowe etapy tego procesu przedstawia rysunek 3.



Rysunek 3 Kluczowe etapy ekstrakcji cech. Źródło: [rozszerzone opracowanie własne na podstawie [53]]

Rysunek 3 prezentuje kompleksowy proces ekstrakcji cech obiektów, który w przypadku analizy tekstu obejmuje: konstruowanie cech, generowanie ich podzbioru, ocenianie otrzymanych podzbiorów oraz zatwierdzanie rezultatów jeśli uprzednio zostało spełnione kryterium stopu. W niektórych zastosowaniach pierwszy etap – konstruowania cech – nazywany jest etapem wstępnego przetwarzania (*ang. preprocessing*). Konstruowanie cech w analizie tekstu zawiera takie działania, jak standaryzacja (*ang. standadization*), normalizacja (*ang. normalization*), wydobycie lokalnych cech (*ang. extraction of local features*) [54]. Dodatkowo, do działań tych można zaliczyć techniki wstępnego przetwarzania dokumentów tekstowych wymienione
w podpunkcie 2.2.1. Konstruowanie cech polega więc na wykorzystaniu całej dostępnej informacji w celu przejścia do nowej przestrzeni. Nowo uzyskana przestrzeń może być, w zależności od wykorzystanych metod, zredukowana, rozszerzona, pozostawiona bez zmian lub wewnętrznie zmieniana w różnych kierunkach. Redukcja wymiaru dotyczy zastosowania metod wbudowanych, które również powodują skonstruowanie nowych cech (pseudo wyrażeń) z cech wyjściowych (podstawowych, bazowych) [55-57]. Transformacja redukująca odbywa się na drodze przekształcenia liniowego bądź nieliniowego. Do liniowych przekształceń należą: analiza składowych głównych (*ang. principal components analysis – PCA)* lub rozkład na wartości osobliwe (*ang. singular value decomposition – SVD*) wykorzystywane w ukrytym indeksowaniu semantycznym (*ang. latent semantic indexing – LSI*) [20, 58]. Natomiast do nieliniowych przekształceń można zaliczyć odwzorowanie Sammona oraz skalowanie wielowymiarowe (*ang. multi dimensional scaling – MDS*) [56]. Rozszerzanie przestrzeni w przypadku analizy tekstu (ekstrakcji wyrażeń) nie znajduje zastosowania. Metodą, która działa i modyfikuje w różnych kierunkach zbiór cech, jest metoda wydobywania cech lokalnych. Przypadek, gdy przestrzeń cech (jej wymiarowość) pozostaje bez zmian, świadczy o zastosowaniu metod z zakresu standaryzacji, normalizacji i zabiegów semantycznych omówionych w podpunkcie 2.2.1.

Etapem, który następuje po konstruowaniu cech obiektów, jest ich selekcja. Polega ona na wyborze możliwie małego podzbioru cech, który da jak największą możliwość rozróżnienia obiektów (dokumentów lub wyrażeń w korpusie lingwistycznym). Należy przy tym zaznaczyć, że może być wiele różnych kryteriów oceny, zależnych od specyficznego zastosowania (zwłaszcza w przypadku podejścia typu wrapper). Wybór cech polega więc na zachowaniu jedynie tych użytecznych, które niosą największą ilość informacji i wyeliminowaniu pozostałych [55]. Proces selekcji z oryginalnego zbioru cech dąży do otrzymania optymalnego ich podzbioru, który zazwyczaj jest niemożliwy do osiągnięcia. Podzbiór ten otrzymywany jest w wyniku procesu (rysunek 3) składającego się z kliku podetapów generowania podzbioru cech na drodze pomiaru i związanego z nim przyjętego kryterium oceny, oraz decyzji czy wygenerowany podzbiór cech jest odpowiedni po spełnieniu zadanego kryterium stopu [53, 54].

Po sparametryzowaniu i wykonaniu etapu generującego podzbiory dochodzi się do ich oceny. Posługując się kryterium oceny podzbiorów, można podzielić algorytmy selekcji cech na cztery kategorie: *filtry*, *wrapper*, *metody wbudowane* (*ang. embedded methods*) oraz *hybrydy* [54, 57, 59].

Przy użyciu wrappera oraz metod wbudowanych można otrzymać różne podzbiory cech z małymi peturbacjami w zbiorze danych. W celu zminimalizowania tego efektu wykorzystuje się zbiór różnych metod (*ang. ensemble learning*) [60]. Dodatkowo, oprócz ww. podziału na filtry, wrappery, metody wbudowane i hybrydy, wprowadzane są kryteria niezależne (*ang. independent criteria*) oraz zależne (*ang. dependent criteria*) [53]. Kryteria niezależne zazwyczaj związane są z modelem filtrów i do oceny podzbioru cech nie wykorzystują żadnego algorytmu eksploracji danych. Kryteria te posługują się pomiarem odległości (*ang. distance measures*), zawartości informacji (*ang. information measures*), zależności (*ang. dependency measures*) i spójności zmiennych (*ang. consistency measures*). Drugie kryterium – zależne,

odnosi się do modelu wrappera i wykorzystuje predefiniowane i wydajne algorytmy eksploracji danych w selekcji cech. Niezależnie od podziału, w przypadku wrapperów oraz rozwiązań hybrydowych, wybrane cechy są dobierane w taki sposób, aby zapewnić możliwe najlepsze wyniki działania docelowej metody (np. grupowanie, klasyfikacja), podczas gdy filtr jest niezależny od stosowanej później metody przetwarzania dokumentów.

Wybór podzbiorów odpowiednich cech i ich ocenianie trwa dopóki nie zostanie spełniony warunek stopu. Rysunek 3 prezentuje ten warunek jako akcję decyzyjną pt. „Kryterium stopu spełnione ?". Warunek stopu jest spełniony gdy spełnione są następujące warunki:
a) przeszukiwanie jest kompletne tj. zbadano całą przestrzeń za pomocą algorytmu przeszukiwania,
b) osiągnięta została specyficzna granica np. ilości iteracji czy też ilości cech,
c) dodawanie lub usuwanie cech nie polepsza i nie generuje ich podzbiorów o lepszych parametrach,
d) określony błąd pomiaru spadł poniżej wyznaczonej granicy.

Ostatnim etapem selekcji cech, choć nie koniecznie kończącym ten proces, jest faza zatwierdzania rezultatów. Bezpośrednio jakość wybranego podzbioru cech można ocenić *a priori* na podstawie jego porównania z cechami jakie się oczekuje. Zazwyczaj taka wiedza *a priori* nie jest dana, wówczas wykorzystywane są metody pośrednie polegające na badaniu jakości osiągów (zwiększanie, bądź zmniejszanie np. celności klasyfikacji) algorytmów eksploracyjnych do wyznaczonego zadania np. klasyfikacji.

W ogólnym przypadku zastosowanie selekcji cech, czy też ogólniej – ekstrakcji cech, ma dodatkowo za zadanie: zredukować dane, zmniejszyć ilość potrzebnej pamięci i tym samym przyczynić się do przyśpieszenia algorytmów operujących na tych danych, zredukować zbiór cech, ulepszyć przetwarzanie (osiągi) związane z dokładnością przewidywania oraz doprowadzić do zrozumienia danych poprzez pozyskanie wiedzy o procesie, który generuje dane i dostarczyć możliwość ich wizualizowania [54].

Koncepcję podziału selekcji cech wyrażoną w postaci trójwymiarowego szkieletu (*ang. three-dimensional framework*), oraz uogólnione, algorytmiczne modele filtrów zostały przedstawione w pracy [53].

2.2.2.2. Wyszukiwanie informacji

Termin wyszukiwanie informacji określa i odnosi się do procesów oraz metod i technik wykorzystywanych w wyszukiwaniu żądanej informacji w zbiorze dokumentów tekstowych) [10, 20, 61, 62]. Wyszukiwanie to odbywa się na podstawie zadanych zapytań składających się z wyrażeń *t* (*ang. terms*). Z dziedziny wyszukiwania informacji wywodzą się też koncepcje dotyczące m.in. budowy i reprezentacji dokumentów tekstowych, ich indeksowania oraz oceny zastosowanego rozwiązania. Koncepcje te stosowane są przy analizach dokumentów tekstowych opisanych w niniejszym opracowaniu.

2.2.2.2.1. Wyszukiwanie informacji – reprezentacja przestrzenno-wektorowa

Na podstawie macierzy *A* z odpowiednio skonstruowanymi wagami w_{ij} możliwe jest wyznaczenie podobieństwa słów oraz dokumentów. Podobieństwo słów wyrażane jest poprzez określenie podobieństwa odpowiadających im kolumn tej macierzy, natomiast o podobieństwie dokumentów wnioskuje się na podstawie analizy podobieństwa wierszy tej macierzy. Najczęściej wszystkie wagi w_{ij} wektorów macierzy *A* w zastosowaniach praktycznych są normalizowane do 1.
W celu określenia miary podobieństwa (dokumentów jak i wyrażeń) stosuje się metryki jak np.: euklidesową, blokową (Manhattanowi), L∞, uogólnioną Minkowskiego L_λ, cosinusową, Jaccarda czy też Dicea [10, 20, 23]. Podobieństwo dokumentów ustala się na podstawie pomiaru odległości. W wyszukiwaniu należy minimalizować odległość maksymalizując w ten sposób podobieństwo. Najpopularniejsze w zastosowaniach metryki, określające podobieństwo dokumentów wrażane są w następujący sposób:
a) miara Euklidesowa, wyrażana jest w postaci wzoru:

$$d_E(i,j) = \left[\sum_{k=1}^{n} (w_k(i) - w_k(j))^2 \right]^{\frac{1}{2}} \quad (4)$$

Gdzie:
- i oraz j – oznaczają *i*-ty i *j*-ty dokument między którymi wyznaczana jest odległość (odpowiednie wiersze macierzy *A* z reprezentacji, którą przedstawia rysunek 2)
- n – ilość składowych (wyrażeń) występujących w macierzy *A*
- $w_k(i)$ i $w_k(j)$ – kolejne *k*-te wagi (wartości obserwacji) dla *i*-tego oraz *j*-tego dokumentu

b) miara Manhattanu (L_1) nazywana także *miarą miejską*, wyrażana jest w postaci wzoru:

$$d_M(i,j) = \sum_{k=1}^{n} |(w_k(i) - w_k(j)) \quad (5)$$

c) miara L∞, wyrażana jest w postaci wzoru:

$$d_{\infty}(i,j) = \max_{k} |(w_k(i) - w_k(j)| \quad (6)$$

d) miara uogólniona Minkowskiego L_{λ}, wyrażana jest w postaci wzoru:

$$d_{\lambda}(i,j) = (\sum_{k=1}^{n} (w_k(i) - w_k(j))^{\lambda})^{\frac{1}{\lambda}} \quad (7)$$

Gdzie:
- $\lambda \geq 1$ – jeśli za λ przyjęte zostanie: $\lambda = 2$ uzyskana zostanie metryka Euklidesowa, $\lambda = 1$ to uzyskana zostanie metryka Manhattanu i $\lambda \rightarrow \infty$ to uzyskana zostanie metryka $L\infty$

e) miara odległości kosinusowa, wyrażana jest w postaci wzoru:

$$d_C(i,j) = \frac{\sum_{k=1}^{n} w_k(i) \cdot w_k(j)}{\sqrt{\sum_{k=1}^{n} w_k(i)^2 \sum_{k=1}^{n} w_k(j)^2}} \quad (8)$$

f) miara Jaccarda, wyrażana jest w postaci wzoru:

$$d_J(i,j) = \frac{2\sum_{k=1}^{n} w_k(i) \cdot w_k(j)}{\sum_{k=1}^{n} w_k(i)^2 + \sum_{k=1}^{n} w_k(j)^2} \quad (9)$$

g) miara współczynnika Dicea, wyrażany jest w postaci wzoru:

$$d_D(i,j) = \frac{2 \cdot |d_i \cap d_j|}{(|d_i| + |d_j|)} \quad (10)$$

Wzór (współczynnik) Dicea można interpretować następująco:

$$d_D(i,j) = \frac{2 \cdot \text{liczba wspolnych wyrazen w dokumencie } d_i \text{ i } d_j}{\text{liczba wyrazen w dokumencie } d_i + \text{liczba wyrazen w dokumencie } d_j} \quad (11)$$

h) miara oszacowania pokrycia (*ang. expected overlap measure*) [23], wykorzystywana gdy wagi wyrażeń w_{ij} zostały wyrażone probabilistycznie (równanie 2). Miara ta wyrażana jest w postaci wzoru:

$$d_{EO}(d_i, d_j, A) = \sum_{t \in d_i \cap d_j} \begin{bmatrix} P(Y_i = t \mid d_i, M) \cdot \\ P(Y_j = t \mid d_j, M) \end{bmatrix} \quad (12)$$

W przestrzeni wektorowej wykorzystując ww. miary podobieństwa istnieje możliwość wyszukiwania dokumentów na podstawie zapytania *Q*. Wyszukiwanie to polega na wnioskowaniu opierającym się na zapytaniu *Q*, prowadzącym do odnalezienia najbardziej podobnych do niego obiektów. Obiekty te w opisywanym przypadku stanowią zbiór dokumentów tekstowych. Zapytanie *Q* może zostać wyrażone w postaci:
a) Boolowskiej funkcji logicznej na zbiorze dostępnych wyrażeń np. *pożar* AND *mocne zadymienie* AND *prąd gaśniczy* AND NOT (*prąd elektryczny*),
b) wektora wag Q = $(q_1,\ldots,q_j)$, gdzie q_j stanowi wagę wyrażenia w zapytaniu i $q_j \epsilon <0,1>$.

Jeżeli zapytanie *Q* będzie składało się z poszukiwanych wyrażeń i w przypadku zastosowania innej reprezentacji ich wag niż Boolowska, to otrzymany zostanie ranking poszukiwanych dokumentów. Zastosowanie zapytania *Q* w wektorowej wagowej postaci wyrażeń i zastosowanie jednolitego zapisu, tj. takiej samej wektorowej reprezentacji dla zbioru dokumentów i wyrażeń w postaci macierzy *A* oraz wektora *Q,* umożliwia stworzenie rankingu poszukiwanych dokumentów. Wyszukiwanie w tym przypadku opiera się na badaniu odległości, która jest określona za pomocą opisanych powyżej miar między wektorem zapytań *Q* składającym się z wybranych wyrażeń i ich wag a macierzą *A* (wierszami w przyjętej
w opracowaniu reprezentacji).

W przypadku zastosowania reprezentacji Boolowskiej zarówno dla *A* jak i *Q* przy wyszukiwaniu nie opartym na mierze lecz na dopasowaniu, istnieje szereg problemów m.in.:
a) brak jest naturalnego znaczenia pojęcia odległości między zapytaniem a dokumentem. W wyniku wyszukiwania uzyskiwany jest nieuporządkowany zbiór (względem miary) dokumentów, pasujących dokładnie do zapytania *Q*,
b) brak jest możliwości wprowadzenia rankingu dokumentów,
c) powstaje problem z konstruowaniem wyrażeń boolowskich, stąd pojawia się problem użyteczności (*ang. usability*) polegający na zrozumieniu przez

użytkownika sposobu formułowania tych wyrażeń i ich stosowaniu.

Mimo tych wad rozwiązanie oparte o reprezentacje Boolowską jest dalej popularne i szeroko stosowane ze względu na implementacyjną prostotę i efektywność. W celu przezwyciężenia ww. problemów stosuje się rozszerzone podejścia boolowskie do reprezentacji i wyszukiwania dokumentów, które pozwalają na uzyskanie rankingu (zakładają one częściowe dopasowanie dokumentów do zapytania). Wykorzystuje się również pozostałe wyżej wymienione odmiany reprezentacji dokumentów tekstowych tj.: częstotliwościową występowania wyrażeń, odwrotną częstość etc. Ich głównym atutem jest to iż umożliwiają tworzenie rankingu istotności zwracanych dokumentów na podstawie zadanego wzorca *Q*.

2.2.2.2.2. Wyszukiwanie informacji – reprezentacja grafowa

W reprezentacji grafowej analogicznie jak w reprezentacji przestrzenno-wektorowej w celu np. wyszukiwania informacji, należy określić czym jest podobieństwo pomiędzy samymi dokumentami jak i dokumentami a wzorcem zapytania *Q*. Wzorzec zapytania *Q* w tym przypadku może być traktowany i reprezentowany jako pewnego rodzaju graf. Schenker i współpracownicy zdefiniowali kilka miar, m.in.: opartych na maksymalnym wspólnym podgrafie, odległości edycyjnej (ile operacji należy wykonać aby przekształcić jeden graf w drugi) itp. Po zdefiniowaniu odpowiednich miar możliwe jest także przeprowadzenie innych niżej opisanych metod analizy tekstu jak np. klasyfikacja czy grupowanie opisanych poniżej.

2.2.2.3. Klasyfikacja dokumentów tekstowych

Klasyfikacja, nazywana także kategoryzacją, dokumentów tekstowych polega na określeniu do jakiej klasy dokumentów można zaliczyć wybrany tekst [18, 63-66] lub jego fragment [67, 68]. Klasyfikacja odbywa się za pomocą wyznaczonego w procesie uczenia klasyfikatora, który będzie dokonywał przyporządkowania dokumentów do jednej lub kilku uprzednio zdefiniowanych klas. Klasy te nie są definiowane wprost, lecz poprzez zbiór trenujący, który stanowi grupa dokumentów już odpowiednio zaklasyfikowana ręcznie np. przez ekspertów. W większości przypadków klasy nie są zagnieżdżane, natomiast przyjmuje się, iż jeden dokument może należeć do więcej niż jednej klasy. Do kategoryzacji dokumentów tekstowych używane są takie techniki, jak: drzewa decyzyjne (*ang. decission tree*), reguły decyzyjne, algorytmy najbliższych sąsiadów i związane z nimi różne metryki (m.in. przedstawione w podpunkcie 2.1.1), klasyfikator bayesowski, sieci neuronowe, metody regresyjne czy też techniki z zakresu maszyn wektorów wspierających (*ang. suport vector machines – SVM*) [69] oraz metody odnajdywania wspólnych podgrafów opartej na metodzie najbliższych sąsiadów ze specjalizowaną miarą odległości, w przypadku zastosowania modelu grafowego dokumentów [30].

2.2.2.4. Grupowanie dokumentów tekstowych

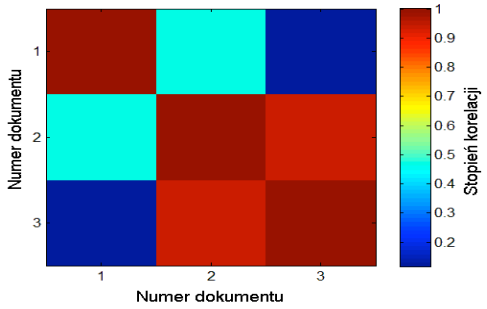
Grupowanie dokumentów tekstowych polega na wyznaczeniu grup podobnych dokumentów np. ze względu na ich tematykę, m.in. za pomocą analizy statystycznej słów występujących w tekście [17, 18, 31, 70-72]. Grupowanie dokumentów tekstowych jest zadaniem pokrewnym do klasyfikacji. W tym przypadku jednak system nie posiada wejściowej wiedzy w postaci już zakwalifikowanych dokumentów, czy też klas wyznaczonych przez ekspertów. Zadaniem tej metody jest takie pogrupowanie dokumentów, by dokumenty należące do jednej klasy były do siebie jak najbardziej podobne i jednocześnie różniły się znacząco od tych należących do innych klas. Do grupowania dokumentów tekstowych używane są takie techniki, jak: analiza skupień, klastrowanie (*ang. clustering*) [73], samoorganizujące się mapy (*ang. self-organization map*) [74], algorytmy aproksymacji wartości oczekiwanej (*ang. expectation-maximization*) [75] czy też zbiory przybliżone [76].

2.3. Wizualizacja

Wizualizacja to metoda związana z końcową realizacją analizy tekstu i wykonywana jest w celu zaprezentowania i lepszego zrozumienia otrzymanych wyników. Głównym celem wizualizacji jest zapewnienie inżynierowi wiedzy lub oprogramowania prostej metody interpretacji uzyskanych wyników. Najczęściej wizualizacji poddawane są związki zachodzące pomiędzy wyodrębnionymi faktami lub zależności zachodzące w strukturze rozpatrywanego zbioru dokumentów tekstowych [34]. Metody wizualizacyjne związane są zarówno z inżynierią wiedzy jak i eksploracyjną analizą tekstu. Do najbardziej znanych metod reprezentacji (wizualizacji) wyników (danych), należą: sieci semantyczne związane z ontologiami, kraty pojęć (*ang. line diagrams*) wykorzystywane w formalnej analizie pojęć (*ang. formal concept analysis – FCA*), histogramy, grafy strony internetowych (*ang. websites as graphs),* wykresy słupkowe, kolumnowe, mapy znaczeń (*ang. mindmaps*), wykresy gwiazdowe, macierze korelacji

narysowane jako obrazy pikselowe wykorzystywane w wyszukiwaniu, klasyfikowaniu oraz grupowaniu dokumentów tekstowych [20, 72, 77-79].

Przykład wizualizacji za pomocą pikselowych macierzy korelacji stosowanych podczas wyszukiwania dokumentów tekstowych. W przypadku gdy dokumenty są reprezentowane za pomocą modelu wektorowego (podpunkt 2.1.1) i gdy jest budowana macierz struktury reprezentacji przestrzenno-wektorowej dokumentów (rysunek 2) o znacznych wymiarach wówczas pomocne okazują się pikselowe macierze korelacji. Ułatwiają one porównanie i wyznaczenie podobnych do siebie dokumentów. Przykładową pikselową macierz korelacji przedstawia rysunek 4.



Rysunek 4 Odległość między parami dokumentów. Źródło: [opracowanie własne]

Rysunek 4 przedstawia sytuację, w której do porównania zostały wzięte trzy dokumenty, tak więc macierz *A* ma wymiary 3x3. Stopień korelacji pomiędzy dokumentami określa się na podstawie wybranej odległości (podpunkt 2.2.2): Minkowskiego, kosinusowej, Jacarda czy też Dicea. Odległość pomiędzy dokumentami w rozpatrywanej macierzy została znormalizowana do 1 i wartościom od 0 do 1 przypisano odpowiednią skalę barw. Kwadraty bliskie niebieskiemu oznaczają dokumenty mniej podobne do siebie, bliskie czerwonemu zaś – bardziej podobne. W przypadku, gdy np. zostanie użyta odległość kosinusowa to wówczas: bardziej czerwone piksele odpowiadać będą większym wartościom kosinusa (bliższe kąty), a bardziej niebieskie dopowiadać mniejszym wartościom kosinusa (większe kąty).

3. PODSUMOWANIE I WNIOSKI

W procesie wstępnego przetwarzania analizy dokumentów tekstowych stosuje się zabiegi związane z automatyczną korektą tekstów (ortograficzną, gramatyczną) w celu polepszenia jakości dokonywanej analizy. Dodatkowo w tym celu stosuje się również takie metody, jak: wykrywanie końca zdań, analizę morfologiczną, usuwanie niejednoznaczności (ekstrakcja rdzeni wyrażeń, lematyzacja), wykrywanie występowania zaimków, wykrywanie nazw własnych i terminów specjalistycznych, rozkład zdań złożonych na zdania proste, rozpoznawanie wyrażeń rzeczownikowych oraz grup czasownikowych, zmniejszanie liter wyrażeń etc. Zadania te należą do głębokiej analizy tekstu. Podczas dokonywania płytkiej analizy tekstu we wstępnym przetwarzaniu zazwyczaj wykorzystuje się tylko część technik z głębokiej analiza tekstu. Rola jej ograniczana jest najczęściej do odfiltrowania zbędnych wyrażeń, znalezienia formy podstawowej wyrażenia lub wyekstrahowania i uwypuklenia najważniejszych poszukiwanych cech w zależności od rodzaju dokonywanej analizy. W dalszym procesie płytkiej analizy pomija się jednak rozpoznawanie wewnętrznej struktury i funkcji wyrażeń w zdaniach czy całych tekstach.

Ze względu na zastosowanie niepełnej głębokiej analizy tekstu na początku procesu płytkiej analizy, otrzymywany jest kompromis w postaci hybrydowego przetwarzania tekstu. W wielu przypadkach np. podczas przeszukiwania i wyszukiwania dokumentów zastosowanie płytkiej analizy tekstu z elementami złożonej analizy we wstępnym przetwarzaniu okazuje się wystarczającym podejściem do uzyskania potrzebnych informacji. Uproszczenia pozwalają na uzyskanie oszczędności czasu w przetwarzaniu dużych korpusów
i grup dokumentów tekstowych. Pomimo ich zastosowania płytka analiza tekstu wciąż jest procesem złożonym i silnie związanym z jakością danych tekstowych (użytego „języka” i jego poprawności do opisu pewnej rzeczywistości) oraz ze słownictwem, które wyznacza kontekst dokumentów np. raporty biznesowe będą posiadać inne słownictwo niż raporty z akcji ratowniczo-gaśniczych. Kontekst ten powoduje iż trzeba będzie poszukiwać i modelować różne zagadnienia
i starać się ekstrahować cechy specyficzne dla danej dziedziny. Powoduje to potrzebę tworzenia narzędzi dedykowanych i profilowanych pod daną dziedzinę zastosowań, nie zaś uniwersalnych, działających na dużym poziomie abstrakcji niezależnym od dziedziny i kontekstu analizy. Oczywiście sam mechanizm takiego wysoko abstrakcyjnego, wstępnego przetwarzania dokumentów tekstowych, może być zaimplementowany. Główny rdzeń płytkiej analizy, prowadzący np. do

wyekstrahowania cech analizowanej dziedziny i przetworzenia wyników w ontologie, już takim automatycznym procesem być nie musi. Wynika to z tego, iż ekspert z danej dziedziny decyduje o tym czy pozyskane atrybuty są przydatne czy też nie w modelowaniu danego zjawiska. Algorytm, czy wybrana technika, jest sama w sobie mało użyteczna w tym sensie, że to człowiek nadaje znaczenie uzyskanym rezultatom w wyniku zastosowania takiego a nie innego podejścia w badaniach.

Należy podkreślić fakt, że wyżej wymienione i opisane metody analizy w zastosowaniach coraz bardziej przestają być autonomiczne. Należy przez to rozumieć, że w celu przeprowadzenia np. wyszukiwania tekstu stosuje się zabiegi związane z grupowaniem dokumentów tekstowych lub grupowaniem pojęć przy wykorzystaniu przykładowo metody ukrytego indeksowania semantycznego [20, 58, 80]. Zabiegi te mają zazwyczaj na celu zmniejszenie, w tym przypadku, przestrzeni wyszukiwanych dokumentów oraz indeksujących je wyrażeń. Poprzez takie mieszane podejście komponowania technik uzyskuje się znaczną poprawę jakości przeprowadzanej analizy.

W przypadku wyszukiwania tekstów następuje polepszenie stosunku dokładności do kompletności w zwracanej odpowiedzi, inaczej mówiąc polepsza się precyzja i przywołanie dokumentów tekstowych na podstawie wygenerowanego zapytania. Możliwe staje się także, w przypadku łączenia wyszukiwania z grupowaniem, otrzymanie wydzielonych grup tematycznych dokumentów w zależności od zadanego wzorca wyszukiwania.

Każda metoda charakteryzuje się własnym sposobem oceniania jakości i dobieraniem odpowiedniej do tego miary. Zagadnienia te są specyficzne i zależne od sposobu wybranej reprezentacji tekstu jak i algorytmu przetwarzania tekstu a nawet samego sposobu indeksowania (różne sposoby indeksowania i metody mogą wpływać np. na szybkość analizy). Wobec tego mierzenie jakości którejś z metod jak i procesu eksploracji dokumentów tekstowych wydaje się być procesem wielowymiarowym i złożonym, zależnym od tego co chcemy osiągnąć w badaniu. Musi jednak pozostać obiektywne, reprezentatywne i krytyczne.

Literatura

[1] Mirończuk M. Eksploracja Danych w kontekscie procesu Knowledge Discovery In Databases (KDD) i metodologii Cross-Industry Standard Process for Data Mining (CRISP-DM). Metody Informatyki Stosowanej, No 2, 2009.

[2] Mirończuk M. Zmodyfikowana analiza FMEA z elementami SFTA w projektowaniu systemu wyszukiwania informacji na temat obiektów hydrotechnicznych w nierelacyjnym katalogowym rejestrze. Studia Informatica, No 2, 2011.

[3] Mirończuk M., Maciak T. Problematyka projektowania modelu hybrydowego systemu wspomagania decyzji dla Państwowej Straży Pożarnej. Zeszyty Naukowe SGSP, No 39, 2009.

[4] Rozporządzenie Ministra Spraw Wewnętrznych i Administracji z dnia 29 grudnia 1999 r. w sprawie szczegółowych zasad organizacji krajowego systemu ratowniczo-gaśniczego. Dz.U.99.111.1311 § 34 pkt. 5 i 6.

[5] Abakus: System EWID99. [on-line] [dostęp: 1 maja 2009] Dostępny w Internecie: http://www.ewid.pl/?set=rozw_ewid&gr=roz.

[6] Abakus: System EWIDSTAT. [on-line] [dostęp: 1 maja 2009] Dostępny w Internecie: http://www.ewid.pl/?set=ewidstat&gr=prod.

[7] Strona firmy abakus. [on-line] [dostęp: 1 marca 2009] Dostępny w Internecie: http://www.ewid.pl/?set=main&gr=aba.

[8] Krasuski A., Kreńsk K. Ewid 9x i co dalej ? Przegląd Pożarniczy, No 6, 2006.

[9] Mirończuk M. Przegląd i klasyfikacja zastosowań, metod oraz technik eksploracji danych. Studia i Materiały Informatyki Stosowanej SIMIS, No 2, 2010.

[10] Mykowiecka A. Inżynieria lingwistyczna. Komputerowe przetwarzanie tekstów w języku naturalnym. Warszawa: PJWSTK, 2007.

[11] Przepiórkowski A. Techniki dezambiguacji morfo syntaktycznej. Powierzchniowe przetwarzanie języka polskiego. Warszawa: Akademicka oficyna wydawnicza EXIT, 2008. s. 17-45.

[12] Vetulani Z. Komunikacja człowieka z maszyną. Komputerowe modelowanie kompetencji językowej. Warszawa: Akademicka Oficyna Wydawnicza Exit, 2004.

[13] Przepiórkowski A., Kupść A., Marciniak M., Mykowiecka A. Formalny opis języka polskiego. Teoria i implementacja. Warszawa: Akademicka Ofcyna Wydawnicza Exit, 2002.

[14] Lubaszewski W. (redaktor) Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu. Kraków: AGH, 2009.

[15] Feldman R., Dagan I., Hirsh H. Mining Text Using Keyword Distributions. Journal of Intelligent Information Systems, No 10, 1998.

[16] Witten I. H., Don K. J., Dewsnip M., Tablan V. Text mining in a digital library. International Journal on Digital Libraries, No 4, 2004, s. 56-59.

[17] Kozłowski J., Neuman Ł. Wspomaganie wyszukiwania dokumentów mapami samoorganizującymi. [Wrocław]: III Krajowa Konferencja MISSI 2002, 19-20 września - „Multimedialne i Sieciowe Systemy Informacyjne", 2002. [dostęp: 10 czerwca 2009] Dostępny w Internecie: http://www.zsi.pwr.wroc.pl/zsi/missi2002/pdf/s507.pdf.
[18] Borycki Ł., Sołdacki P. Automatyczna klasyfikacja tekstów. [Wrocław]: III Krajowa Konferencja MISSI 2002, 19-20 września - „Multimedialne i Sieciowe Systemy Informacyjne", 2002. [dostęp: 10 czerwca 2009] Dostępny w Internecie: http://www.zsi.pwr.wroc.pl/zsi/missi2002/pdf/s504.pdf.
[19] Neumann G., Piskorski J. A Shallow Text Processing Core Engine. Computational Intelligence, No 18, 2002, s. 451-476.
[20] Hand D.,, Mannila H., Smith P. Eksploracja danych. Wydanie 1. Warszawa: Wydawnictwo Naukowo-Techniczne, 2005.
[21] Morzy M., Królikowski Z. Metody indeksowania atrybutów zawierajacych zbiory. Pro Dialog, No 15, 2003, s. 87-106.
[22] Dudczak A. Zastosowanie wybranych metod eksploracji danych do tworzenia streszczeń tekstów prasowych dla języka polskiego. Wydział Informatyki i Zarządzania Instytut Informatyki. Poznań: Politechnika Poznańska 2007.
[23] Goldszmidt M., Sahami M. A Probabilistic Approach to Full-Text Document Clustering. 1998.
[24] Singhal A., Buckley C., Mitra M., Mitra A. Pivoted Document Length Normalization. ACM Press, 1996, s. 21-29.
[25] Robertson S. E., Walker S., Jones S., Hancock-Beaulieu M. M., Gatford M. Okapi at TREC-3. 1996, s. 109-126.
[26] Lin D. Using syntactic dependency as local context to resolve word sense ambiguity. [Madrid, Spain]: Annual Meeting of the ACL Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, 1997.
[27] Matsuo Y., Ishizuka M. Keyword Extraction From A Single Document Using Word Co-Occurrence Statistical Information. International Journal on Artificial Intelligence Tools, No 13, 2004, s. 157-169.
[28] Maciołek P., Dobrowolski G. Propozycja metody klasyfikacji dokumentów w języku polskim. In: Grzech A., Juszczyszyn K., Kwaśnicka H. and Nguyes N.T., editors. Inżynieria wiedzy i systemy ekspertowe. Warszawa: Akademicka oficyna wydawnicza EXIT, 2009.
[29] Chow T. W. S., Haijun Zhang, Rahman M. K. M. A new document representation using term frequency and vectorized graph connectionists with application to document retrieval. Expert Systems with Applications, No 36, 2009, s. 12023-12035.
[30] Schenker A., Kandel A., Bunke H., Last M. Graph-Theoretic Techniques for Web Content Mining. World Scientific Publishing Co, 2005.
[31] Broda B. Mechanizmy grupowania dokumentów w automatycznej ekstrakcji sieci semantycznych dla języka polskiego. Wydział Informatyki i Zarządzania. Wrocław: Politechnika Wrocławska, 2007.
[32] Gruber T. R. A translation approach to portable ontology specifications. Knowledge Acquisition, No 5, 1993, s. 199-220.
[33] Meyer B. Programowanie zorientowane obiektowo 2005.
[34] Lula P. Text mining jako narzędzie pozyskiwania informacji z dokumentów tekstowych. StatSoft, 2005.
[35] Savinov A. Concept-Oriented Model. In: Ferraggine V. E., Doorn J. H., Rivero L. C., editors. Handbook of Research on Innovations in Database Technologies and Applications: Current and Future Trends: IGI Global, 2009.
[36] Savinov A. Principles of the Concept-Oriented Data Model. 2004. [dostęp: 22 grudnia 2009] Dostępny w Internecie: http://conceptoriented.com/savinov/publicat/imi-report'04.pdf.
[37] Savinov A. Informal introduction into the Concept-Oriented Data Model. 2005. [dostęp: 22 grudnia 2009] Dostępny w Internecie: http://conceptoriented.org/papers/ComInformalIntroduction.pdf.
[38] Savinov AA. Concept-Oriented Model and Query Language. CoRR, No abs/0901.2224, 2009.
[39] Praca zbiorowa Wikipedia Full text search. [dostęp: 22 grudnia 2009] Dostępny w Internecie: http://en.wikipedia.org/wiki/Full_text_search.
[40] Moens M. F. Information Extraction: Algorithms and Prospects in a Retrieval Context (The Information Retrieval Series). Springer, 2006.
[41] Bikel D. M., Schwartz R., Weischedel R. M. An Algorithm that Learns What's in a Name. Machne Learning, 1999, s. 211-231.
[42] McNamee P. Language identification: a solved problem suitable for undergraduate instruction. Journal of Computing Sciences in Colleges, No 20, 2005, s. 94 -101

[43] He X., Yang M., Gao J., Nguyen P., Moore R. Improved Monolingual Hypothesis Alignment for Machine Translation System Combination. No 8, 2009, s. 1-19.
[44] Feng Y., Liu Y., Mi H., Liu Q. Lattice-based system combination for statistical machine translation. [Singapore]: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Volume 3, 2009.
[45] He X., Toutanova K. Joint optimization for machine translation system combination. [Singapore]: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Volume 3, 2009.
[46] Afantenos S., Karkaletsis V., Stamatopoulos P. Summarization from medical documents: a survey. No 33, 2005, s. 157-177.
[47] Turney P. D. Learning Algorithms for Keyphrase Extraction. Information retrieval, No 2, 2000, s. 303-336.
[48] Turney P. D. Mining the Web for Lexical Knowledge to Improve Keyphrase Extraction: Learning from Labeled and Unlabeled Data. 2002.
[49] Indyka-Piasecka A. Model użytkownika w internetowych systemach wyszukiwania informacji Wydział Informatyki i Zarządzania. Wrocław: Politechnika Wrocławska, 2004.
[50] Dasgupta A., Drineas P., Harb B., Josifovski V., Mahoney M. W. Feature selection methods for text classification. [San Jose, California, USA]: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, 2007.
[51] Li S., Xia R., Zong C., Huang C. R. A framework of feature selection methods for text categorization. [Suntec, Singapore]: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. Volume 2, 2009.
[52] Karlgren J., Sahlgren M. From Words to Understanding. 2001. [dostęp: 10 stycznia 2010] Dostępny w Internecie: http://www.sics.se/~mange/papers/KarlgrenSahlgren2001.pdf.
[53] Liu H., Yu L. Toward integrating feature selection algorithms for classification and clustering. Knowledge and Data Engineering, IEEE Transactions on, No 17, 2005, s. 491-502.
[54] Guyon I., Elisseeff A. Introduction to Feature Extraction. Studies in Fuzziness and Soft Computing. Berlin/Heidelberg: Springer 2006.
[55] Torkkola K. Feature extraction by non parametric mutual information maximization. The Journal of Machine Learning Research, No 3, 2003, s. 1415-1438
[56] Pal S. K., Mitra P. Pattern Recognition Algorithms for Data Mining Scalability, Knowledge Discovery and Soft Granular Computing. London New York Washington, D.C.: CHAPMAN & HALL/CRC, 2004.
[57] Praca zbiorowa JMLR Special Issue on Variable and Feature Selection. [dostęp: 5 stycznia 2010] Dostępny w Internecie: http://jmlr.csail.mit.edu/papers/special/feature03.html.
[58] Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K., Harshman R. Indexing by latent semantic analysis. Journal of the American Society for Information Science, No 41, 1990, s. 391-407.
[59] Kozłowski M. Systemy uczące się - studium problemów. Warszawa: Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych. [dostęp: 12 stycznia 2010] Dostępny w Internecie: http://home.elka.pw.edu.pl/~mkozlow3/artykuly/M.Kozlowski.pdf.
[60] Tuv E. Ensemble Learning. In: Guyon I., Gunn S., Nikravesh M., Zadeh L. A., editors. Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing) (Hardcover): Springer, 2006.
[61] Baeza-Yates R., Ribeiro-Neto B. Modern Information Retrieval. Boston: Addison-Wesley Longman Publishing, 1999.
[62] Manning C. D., Raghavan P., Schtze H. Introduction to Information Retrieval. Cambridge University Press India, 2008.
[63] Song F., Liu S., Yang J. A comparative study on text representation schemes in text categorization. Pattern Analysis & Applications, No 8, 2005, s. 199-209
[64] Weigend A. S., Wiener E. D., Pedersen J. O. Exploiting Hierarchy in Text Categorization. Information Retrieval, No 1, 1999.
[65] Yang Y., Liu X. A re-examination of text categorization methods. [New York]: ACM SIGIR Conference of Research and Development in Information Retrieval, 1998.
[66] Łażewski Ł., Pikuła M., Siemion A., Szklarzewski M. Klasyfikacja dokumentów tekstowych. Warszawa: PJWSTK 2005. Dostępny w Internecie: http://www.scribd.com/doc/2242106/Klasyfikacja-dokumentow-tekstowych.
[67] Agarwal S., Yu H. Automatically classifying sentences in full-text biomedical articles into Introduction, Methods, Results and Discussion. Bioinformatics, No 25, 2009, s. 3174-3180.
[68] Sebastiani F. Machine learning in automated text categorization. ACM Comput Surv, No 34, 2002, s. 1-47.
[69] Aas K., Eikvil L. Text Categorisation: A Survey. Technical Report, Norwegian Computing Center, 1999.

[70] Weiss S., White B., Apte C., Weiss S. M., White B. F., Apte V. Lightweight Document Clustering. 2000.
[71] Domeniconi C., Gunopulos D., Ma S., Papadopoulos D., Yan B. Locally adaptive metrics for clustering high dimensional data. Data Mining and Knowledge Discovery, No 1, 2006, s. 63-97.
[72] Solka J. L. Text Data Mining: Theory and Methods. Statistic Survey.
[73] Everitt B. S., Landau S., Leese M. Cluster Analysis. 2001.
[74] Kohonen T. Self-Organizing Maps. In: Sciences S.S.i.I., editor. Wydanie 3. Berlin: Springer, 2001.
[75] Dempster A. P., Laird N. M., Rdin D. B. Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society, No 39, 1977, s. 1-38.
[76] Rutkowski L. Metody i techniki sztucznej inteligencji. Wydawnictwo Naukowe PWN, 2005.
[77] Wolff K. E. A first course in formal concept analysis. 1994. [dostęp: 22 grudnia 2009] Dostępny w Internecie: http://www.fbmn.fh-darmstadt.de/home/wolff/Publikationen/A_First_Course_in_Formal_Concept_Analysis.pdf.
[78] Friedman V. Data Visualization: Modern Approaches. [dostęp: 29 grudnia 2009] Dostępny w Internecie: http://www.smashingmagazine.com/2007/08/02/data-visualization-modern-approaches/.
[79] Piwowar K. Wizualizacja danych a ich używalność – czyli pokazać to tak, aby inni to zrozumieli. [dostęp: 29 grudnia 2009] Dostępny w Internecie: http://interaktywnie.com/biznes/blog-ekspercki/blogi/wizualizacja-danych-a-ich-uzywalnosc-8211-czyli-pokazac-to-tak-aby-inni-to-zrozumieli-384.
[80] Osiński S., Weiss D. Projekt „Lingo" i Carrot2. [dostęp: 1 stycznia 2010] Dostępny w Internecie: http://carrot.cs.put.poznan.pl/stable/search.

Projekt współfinansowany ze środków Europejskiego Funduszu Społecznego w ramach Programu Operacyjnego Kapitał Ludzki Działanie 8.2 Transfer wiedzy, Poddziałanie 8.2.2 Regionalne strategie innowacji, budżetu państwa oraz środków Samorządu Województwa Podlaskiego.