Automated Acquisition of Semantic Knowledge to Improve Efficiency of Information Retrieval Systems

Dariusz Ceglarek, Wojciech Rutkowski
The Poznan University of Economics
Department of Management Information Systems
{d.ceglarek,w.rutkowski}@kie.ae.poznan.pl

Abstract

This paper presents the idea of incorporating semantic knowledge into information retrieval systems is presented. Semantic knowledge can be represented by structures such as thesauri or semantic networks. These structures are, however, very extensive and their manual construction is a complex and time-consuming task. That is why several methods allowing to acquire semantic relations in an automated way are being presented.

1. Introduction

Considering a continuous rise of world's information resources, it is necessary for companies and other organizations to obtain, aggregate, process and utilize them in an appropriate manner in order to maximize the efficiency of activities that are being conducted.

Since the document libraries (or a number of sources to filter from) are becoming bigger and bigger, it is crucial to provide a trusted system which would be able to find the resources relevant to user's needs. This is a main goal of information retrieval (IR) systems. [Daconta 2003]

Traditionally, the efficiency of IR systems is measured by two basic factors: *recall* and *precision*. Both are quantified by a percentage or a value between 0 and 1. Suppose we have a set of documents. A user has specific information needs, represented by a *query*. Task of IR system is to provide the user with *relevant* documents from the set. Recall equals a relation of relevant documents returned by IR system to the number of all relevant documents, and precision is a relation of returned relevant documents to all returned documents.

Were everything perfect, the recall and precision of IR system would reach 100%. This is a goal of developing and improving retrieval systems.

1.1. Economic Significance of Recall and Precision

In business efficiency of retrieval systems reflects on profitability. The higher recall the system achieves, the more relevant and, therefore, valuable documents are delivered. In other words, full recall means the widest access to all useful information.

The second indicator, precision, affects the time resources needed to browse and utilize the results. In case of low precision of retrieval systems, lots of manual work is required to determine which of the results are useful (relevant to the needs) and reject inessential (irrelevant) resources. The higher precision, the more time is spared on such process, what directly reflects on work efficiency and costs.

Concluding, improving the efficiency of retrieval systems is beneficial for organizations whose activities are based on information and knowledge.

1.2. Topic of Interest

Information searching and exploring takes place in a domain dependent semantic context. A given context is described through its vocabulary organized along hierarchies that structure the information space. These hierarchies are simplified views on a more complex domain specific semantic network, that form a shared and coherent background knowledge representation. The exploration of documents is more effective. Hierarchies (extracted from the semantic networks) provide with a language and synthetic representation to be explored by the users to express their information need.

This paper shows that improving the efficiency of information retrieval by utilizing more and more sophisticated models grooves on inclusion of mechanisms reflecting and using information of semantic relations between concepts of language.

Text documents are most often a subject of retrieval, the complexity of human natural language, however, negatively affects the results of classic algorithms implemented in IR system.

Retrieval methods are getting accommodated to identify semantic relations between word (i.e. relations between the meanings of words or the concepts they represent) and use this knowledge to compare and match documents with user's needs more accurately. Such knowledge is represented in structures like thesauri or semantic networks.

Thesauri or semantic networks can be created manually Unfortunately, it is a very time-consuming task and needs an involvement of expert knowledge. This paper reviews methods which allows to automatically extract mentioned knowledge – semantic relations between words – from a corpus of documents.

The above approach is supported by following arguments. Utilizing semantic relations improves the efficiency of IR systems: they allow to increase the recall of retrieval by identifying potentially more relevant documents, and refining ranking functions has an impact on higher retrieval precision. Economic significance of such improvement has been described in previous subsection.

In turn, automated acquisition of semantic relations means a huge facilitation in creating structures representing language knowledge (as for example thesauri) which can be used in retrieval systems.

Special emphasis has been put on usefulness of presented methods in retrieval systems for flexile languages, especially Polish. High flexibility of such languages, meaning a multiplicity of word forms, is an additional difficulty for methods or algorithms which performance is satisfactory when working with less flexible languages.

1.3. Methodology

This paper is based on review of recent research and publications concerning especially the last five years. Within the author's research, some experiments were conducted including implementation of proposed methods and evaluation of results.

Evaluation was done on a corpus of 4 thousand documents, articles gathered from a news portal. Every document had its structure analyzed, the text was tokenized and the resulting words lemmatized. The procedure is described in detail in section 4.2.

1.4. Paper Structure

The paper is divided into five sections. The introduction presents the problem mentioned in this paper, proposes a solution, and suggests some economic benefits which can be achieved by using the solution.

The second section is a review of information retrieval system models. It indicates that development of IR grooves on incorporation of semantic knowledge to retrieval methods.

The third section describes semantic relations and structures which can be used to represent language knowledge to use in retrieval systems.

The fourth section introduces methods allowing to automatic acquisition and gathering information about semantic relations in text documents. It also contains an evaluation of proposed methods.

The last section summarizes the paper and outlines findings, that were discussed in previous parts.

2. Evolution of Information Retrieval Systems

Information retrieval may be characterized as a system which provide the user with documents that will best satisfy their need for information. Different approaches have been proposed in the literature to enhance system effectiveness, specifically methods to improve the document representation or matching documents with a query, for instance by query reformulation.

With regard to document representation which is a key point in IR a common solution is to choose significant sets of weighted terms. Several works have investigated a richer representation in order to get better query matching. Natural Language Processing (NLP) is one of the means that have been tested. An alternative way to go beyond "bags of words" could be to organize indexing terms into a more complex structure such as a hierarchy or an ontology. Texts would be indexed by concepts that reflect their meaning rather than words considered as chart lists with all the ambiguity that they convey.

In this section some information retrieval models have been presented with an aim to understand their evolution and emphasise a tendency of incorporating word semantics into retrieval algorithms. [Baeza-Yates 1999]

A *Boolean model* is the most basic retrieval model. It is operated on set theory: set of documents returned by retrieval system is a conjunction of sets representing respective query words. Semantic knowledge, thus, cannot be used in Boolean model at all. Documents from library can either match the query or not – returned documents are not ranked or sorted in any way. Therefore, the Boolean model is the simplest, and on the other hand, the most primitive one.

A *ranking function* which orders a collection of documents by their probable relevance to a user query is introduced in *algebraic models* such as *vector space model* (VSM). Vector models are very common and create a base for further development.

In VSM particular words are represented as dimensions of space. Documents are represented by vectors in this space. Size of the vectors is determined by an occurrence frequency of respective words in the document. User query is represented as a vector as well. Matching documents to query or comparing a similarity of two documents is done by comparing their vectors.

Vectors can be compared via algebraic methods and one of them is a cosine measure. It depends on the angle between vectors – the closer the vectors are, the smaller angle. Cosine measure has a continuous value between -1 and 1 – the maximum value when two compared vectors are collinear.

Advantage of the VSM is the continuous ranking function which allows to fluent affection on precision and recall of IR system. Depending on assumed threshold, retrieval system can return more documents (what lifts up the recall but lowers the precision) or less documents (what means lower recall but higher precision). Therefore, recall and precision indicators, as well as the number of returned documents, can be easily controlled.

The main disadvantage of vector models is the presumption that the dimensions are orthogonal, that is the words are fully independent. Obviously, this is not the case when natural language is concerned.

The above restriction induces further evolution of IR. Alternative retrieval models have been developed as improvements of classical models. The main tendency observed in the alternative models involves possible interdependencies between terms and register them as semantic relations between concepts. That was not a case in basic models.

Fuzzy logic model operates on fuzzy sets theory. The theory of fuzzy set with definition of operations on such sets was described by Lotfi A. Zadeh in 1965. The theory rejects the restriction that an element must either belong to a given set or not. Fuzzy set theory permits the gradual assessment of the membership of elements. Every element has a *membership function* which determines its degree of membership to a given set.

Fuzzy logic IR model rejects the constraint of Boolean logic model where term frequency in a documents has no influence on ranking function, that means that it does not matter whether a word is frequent in a document (and, therefore, semantically more important) or it occurs just one time [Zadeh 1965].

Another alternative retrieval model is *neural network* model. It uses a neural network with three layers: query layer (consisting of neurons representing query words), dictionary layer (built from neurons representing all processed words) and document corpus layer (one neuron represents one document). There are links between the second and third layer with weights. The weight represents frequency of the word occurring in the document.

In neural network retrieval model signal form query terms is sent to second layer neurons which became activated. Then, modified by appropriate weights, signal passes to document where it accumulates. Activated documents can be returned or backward propagation can be processed.

The backward propagation emulates hidden semantics which is not represented by simple term to document frequency. Especially the terms activated as a result of backward propagation can be semantically similar to query terms but are not explicitly included in the query. This effect causes a rise in retrieval recall and, providing good weighting, more precise result ranking.

Another retrieval model concerning hidden semantics is *Latent Semantic Indexing* (LSI). It is based on the vector space model but reduces the number of dimensions in order to improve speed and efficiency. Its main task is to limit the number of dimensions by representing the documents as good as possible. In LSI term frequencies and co-occurrences are analyzed and the most similar words are merged into synthetic concepts. Therefore, the number of dimensions is being reduced, too. [Letsche 1996]

In the above short preview of retrieval models we can observe two different trends. The first is a tendency to improve a ranking function – it needs to be continuous and allow to control the efficiency of IR system fluently by exchanging recall and precision.

In the above evolution of IR it can be seen that the more sophisticated and effective model is, the more semantic knowledge it utilizes [Hotho 2003]. Particular models entertain relations between words, their co-occurrences and similarity. User queries can be expanded by lexically related words [Mandala 1999].

That is why an idea of including semantic knowledge represented by such structures as thesauri or semantic networks is suggested. In the next section we will describe semantic knowledge representation. In section 5, on the other hand, several methods allowing to create these structures in an automated way are reviewed.

3. Representation of Semantic Knowledge

According to the introductory statement, natural language is a very complex system which needs to be represented in a way that would be understandable for computer systems. In this section some structures that can represent a part of semantic knowledge have been described.

From data processing point of view, words are strings of characters. These strings can be compared, and as a result – equal or different. There are some metrics characterizing the similarity between the strings of characters, like the Levenshtein distance which says how many characters have to be deleted, inserted or substituted to transform one string into another. But such indicators do not say about the similarity of concepts represented by words and do not, thus, reflect the complexity of language.

Some of main relations identified in natural language are described in the following subsection.

3.1. Word Relations

Two groups of word relations can be distinguished: collocations and lexical relations. The former is based on word co-occurrences and connection with their common meaning while the latter affects the concepts represented by words.

Collocation is a pair of words often occurring together in a text. The meaning of the word pair results from a sum of meanings or can be totally different as in the case of idiomatic expression. Collocations are statistical phenomenon which can be observed using statistical methods, and is tightly connected with word meanings.

Lexical relations between words reflect the interdependences between the concepts – meanings of the words. The most important lexical relations are described below.

Synonymy and *similarity* are the relations occurring between words with corresponding concepts that are equal or close near, respectively. Similarity can be graded as discrete or continuous value.

Antonymy is a relation occuring when word meanings are opposite.

Meronymy and *holonymy* occurs between words when concept represented by the first word is a part of a concept represented by the second word. *Meronym* is the word that represents a part of *holonym* – word that possesses the part.

Hypernymy and *hyponymy* express hierarchical relations between concepts. Hypernym is a word with broader meaning than a narrower hyponym. In other words, a concept of the hypernym is a superordinate of a hyponym's concept.

As in the case of similarity, the above relations can be weighted – their strength can be graded as a value, either discrete or continuous.

Identifying semantic relations, assembling information about the relations, and utilizing it to refine retrieval systems, streamlines the results of IR system, as stated before, has a rational impact on efficiency of activities based on appropriate information set.

3.2. Structures

Semantic knowledge, as identified semantic relations between words, should be stored in an appropriate data structure in order to be utilized to refine retrieval systems and their results.

Dictionary is a structure to start with. It does not contain the information about semantic relations but is only a vast set of words that serves as a base for further processing. Dictionary is a representation of words occurring in considered collection of text documents (document corpus). Depending on application, we can distinguish several types of dictionaries:

- dictionary of all words occurring in document corpus,
- dictionary of words typical for a given topic,
- dictionary of words of specific part of speech,
- defining dictionary (one-language dictionary),
- two-language dictionary (or multi-language),
- stoplist.

A particular dictionary is a frequency dictionary, which is a structure containing information about number or frequency of given words in document corpora. All kinds of dictionaries previously mentioned can be useful in text processing or acquiring semantic relations, what is described in section 4.

Structures described below include information of semantic relations in addition to the set of words. Some concepts from graph theory are used to characterize these structures. *Graphs* are ordered sets of *nodes* joined by *edges*. In particular, we use a concept of *directed graph*, where every link between two objects has a specified direction (such edge is called an *arc*), a *tree* – directed, coherent graph without cycles, and a *rooted tree* – a tree with one node distinguished as a root. Graphs, therefore, can represent semantic knowledge quite well: concepts are represented by graph nodes and arcs are semantic relations between concepts. [Cormen 2001]

Taxonomy is a structure representing hierarchical relations between concepts or corresponding words. Taxonomies are common in sciences, used to organize domain terminology. Semantic relations included in taxonomy are hypernymy and hyponymy. In case of two concepts linked with such relations, the higher one in the hierarchy is a superordinate term for the second one (subordinate term).

Using graph theory terminology, a taxonomy is a rooted tree with distinguished root concept. The further from the root words are, the more specific concepts they represent. Analogically, the shorter path from root concept to a word is, the more general the word is.

A structure containing word similarity relations is called a *thesaurus*. Thesauri incorporate such lexical relations as synonymy and antonymy – as particular, opposite sorts of word similarity (identity in case of synonymy, reverse in case of antonymy). Thesauri are included into IRS mechanisms while they are getting more advanced and efficient, mainly to expand user queries, match and compare documents in a better way [Jing 1994].

Finally, a *semantic network* is a structure incorporating knowledge about all possible semantic relations between words. Semantic networks store information about similarity relations (like a thesaurus): word similarity, synonymy, antonymy; hierarchical relations (like a taxonomy): hypernymy or hyponymy and meronymy or holonymy relations. Semantic network can incorporate connotations as well – these are any other word associations.

Using the graph theory terminology, semantic networks can be represented as directed graphs. Direction is crucial in case of hierarchical relations. Edges between concepts can be weighted as well – in order to reflect strength of a relation.

Semantic networks are the most advanced structures representing semantic knowledge. That is why their utilization in information retrieval systems should bring the biggest improvement in their efficiency. The information included in semantic network can be used in order to limit the number of keywords to describe a document, expand user queries or identify concepts if a word represents more than one meanings.

4. Automated Acquisition of Semantic Relations

Use of information on semantic relations leads to an improvement of IR efficiency. The more relations in a structure, the better usefulness for retrieval system. Building an extensive semantic network (or other structure) is, however, a complex and time consuming task.

Dictionary of domain keywords contains usually hundreds thousand words (biggest one – few millions of words). Then, relations are to be added: several links for each word resulting with thousands, or even millions relations.

Creating such a huge structure needs a lot of work and is nearly impossible. That is why automated methods can help – by acquiring semantic relations in textual documents and incorporating them into a structure as, for instance, semantic network.

In this section several methods allowing to automatically acquire semantic relations betweens words in a corpus of documents are being presented.

Some preconditions have been assumed. First, all documents are written in one language. Originally, most of methods were developed and evaluated for the English language. For purpose of this work presented methods were implemented and evaluated for the Polish language exemplifying one of highly flexible languages and generating some additional morphological problems. Second, the corpus of textual documents consists of documents concerning one domain. This condition limits the effect of homonymy – different meaning of one word. It is quite helpful and realistic since document corpora in business concern mainly one domain. On the other hand, there are methods allowing to specify a meaning of a word representing multiple concepts [Schuetze 1995].

4.1. Flexible Languages

Problems emerge when documents are written in highly flexible language. They are characterized by a multiplicity of forms of one word and complicated syntax.

In languages with simple word inflection (such as English) multiple forms of one word can be reduced to one term using *stemming algorithms*. The idea of such algorithms in mainly cropping word suffix, following defined rules. Stemming is relatively simple when the returned string is not necessarily a right word, only an unique identifier. A task of providing a word in basic form for a given word in any form is lemmatization.

Let's look at the inflection of the verb "lock" in order to compare the complexity of lemmatization task for English and highly flexible languages. In English, it has four forms: *lock, locks, locking, locked*. In Polish, the same word is "blokować", and the inflection of this word has 37 forms.

There are two ways of constructing a lemmatizing tool: dictionary or rule approach. The former is to build a dictionary of all word forms with links to their basic forms (*lemmas*). The latter is to define a set of rules by which word forms are reduced to their lemmas. The first approach is accurate providing that all words are present in the directory. The second approach finds lemma for every word, nevertheless, there is a possibility that the returned lemma is not a correct word.

The solution which joins the benefits of dictionary and algorithmic lemmatizer is a *hybrid lemmatizer* [Weiss 2005]. Every given word is checked in a dictionary and, if any appropriate entry exists, lemma is returned. If there is no such word, rule methods are executed and lemma is derived.

Hybrid lemmatizer can find a lemma for every given word with small chance of returning wrong word.

4.2. Methods

In this subsection several methods allowing to build a structure representing semantic knowledge are being described. As stated before, first step is to gather an appropriate set of words from document corpus representing a given domain.

Every document has its structure analyzed. Chapters, sections, and paragraphs are being detected at this point. Then, lexical analysis is executed – processing the text and splitting it into particular words. Word are split by detecting word separators – that is spaces, punctuation. Words are subject to lemmatization, to limit the effect of inflection, and *stopwords* (very frequent words without or with very small semantic importance – such as conjunctions) are removed from further analysis. After removing multiple word occurrences the result is a set of unique words in the whole corpus. [Frakes 1992]

In algorithmic lemmatization tests (using two sets of words, counting over 70,000 and 130,000 words) the number of unique words (in many forms) was reduced by 65-75%, and most of the unrecognized words (about 15%) were in foreign language, erroneous or proper names.

At the phase of lexical analysis *collocations* can be detected. Collocation is a relationship between words that often occur together forming an expression and common concept. Automated acquisition of collocation is processed with statistical methods – every pair (or group) of words is counted and the resulting number is compared with frequencies of single words' occurrences. Is the relation of pair frequency to single words frequencies high, the pair is considered as a collocation. [Evert 2001]

If a user query contains a collocation it should trigger searching the same collocation (with respect to word order) in documents in retrieval process as if the collocation were a single concept.

In a corpus of documents of a 2 million words size as an effect of computing frequencies of every pair of words, there were nearly 3,000 collocations occurring more than 10 times. The efficiency of finding a correct collocation was estimated 85%.

At this phase the set of words can be reduced to domain dictionary. This can be achieved by comparing word frequencies in two document corpora: one general and one domain. Words with higher frequency in domain corpus than in general corpus can be considered to be typical for the domain while words with similar frequencies are words used commonly in everyday language.

Main advantage of the above method is limiting the size of base dictionary and – in case of VSM – reducing the number of dimensions, which improves the algorithm speed.

Building domain dictionary by comparing word frequencies is very effective. In evaluation, an input dictionary (20,000 lemmas) was reduced close to 3,000 words typical to the domain given (and, therefore, most valuable for retrieval purposes) thanks to selecting words which relative frequency in domain document corpus (in comparison to frequency in general corpus) was higher than average.

Having a domain dictionary we can start acquiring semantic relations. First method, allowing to automatically acquire similarity relations, depends on word co-occurrences. The idea of this method, is to compute the correlation between occurrences of two words in one block of text (paragraph, section, sentence – provided by structure analysis).

A simple solution is used in LSI model which depends on computation of Pearson's correlation coefficient. More sophisticated approach assumes building a vector for every word. Here the dimensions represent documents and the weights correspond to importance of a respective word in a document [Qiu

1995]. Comparing word vectors can be conducted with the same measures as in VSM, that is, for instance, cosine measure.

Another method uses conditional co-occurrence. Correlation coefficients are calculated for a pair of words by measuring frequency of the first word only in documents with the second word present. High conditional co-occurrence for a given pair of words can mean that the first word occurs nearly always when the second word is present. If the measure calculated for the same pair of words with reversed order (that is frequency of the second word in documents containing the first word) is low we can assume that the first word is a broader term than the second. That is, some kind of hierarchical relation between these words, particularly, the first word is a hypernym of the second word which is a hyponym. [Sanderson 1999]

In evaluation, 700 nouns from a given domain were selected and then conditional co-occurrence computed for every pair. Making two additional assumptions, that the probability of conditional co-occurrence must be above 70% and the pair of words must co-occur in minimum 0.5% of documents, over 400 relations were returned as a result. In this group 45% relations were correct pairs of hypernym-hyponym. Another 25% represent other semantic relations. Other relations were erroneous but contained a variety of proper names. It was estimated that by automated identification of proper names the efficiency of hierarchic relation acquisition method could reach 60-70%.

In order to automatically acquire synonymy relations the method based on a two-language directory has been proposed. It uses a feature of such dictionary to propose more than one translation to a given meaning. As a result, there are few words – synonyms – collected around one concept.

OpenThesaurus project is an implementation of the method [Naber 2004], and according to the developers, its accuracy reaches 90% while the rest is being corrected manually.

There is one more method allowing to automatically derive hierarchic relations. Similarly to the previous method, it uses dictionary (a one language, definitional dictionary in this case). This method builds *definitional sequences* – starting from any word (usually noun) it gets a definition of the word. The definition is then parsed and lexically analyzed to find the first noun in the definition. The method is based on an observation that the first noun in a definition of word is usually its hypernym [Hammerl 1993].

Basing on the above property and making use of definitional dictionary, several semantic relations can be acquired in an automated way. As an evaluation, 700 words were used to determine their hypernyms automatically. The assumption that in case of multiple definitions only the first will be used, appeared. The accuracy of this method was estimated to 85%, which is a good result taking the assumption into account.

Acquiring semantic relations of virtually every type is possible using a method of detecting key phrases. It is based on an assumption that two words linked with a semantic relation in the text often occurs in proximity, are separated by a characteristic terms. For example, such phrases as "…, and other

…", "… including …", "… such as …", "… is a part of …" indicate that the word before and the word after are semantically related [Hearst 1992][Koo 2003].

The above method was developed and tested for the English language. However, when evaluated on highly flexible language, it performs poor. Search for 20 key phrases in document corpus resulted in about a hundred pairs of words. Three best phrases were found 60 times, showing correct relations in 50%.

In order to acquire hierarchic relations between terms, *formal concept analysis* was proposed [Cimiano 2003]. It utilizes an observation that most of verbs can be used together with objects having some specified features (for example "flyable", "fluid", "edible", "drivable"). Considering sets of such features, it is possible to build a hierarchy of objects: these, having more features, are broader terms (hypernyms) while the objects with less attributes are narrower terms (hyponyms).

In evaluation of all presented methods a polysemy effect was omitted. The methods, thus, could show higher efficiency than estimated. There is research on identifying concepts represented by homonyms and some methods allowing to determine which of the meanings is represented in a given document have been developed. The methods base mainly on context analysis – identifying characteristic, typical to respective meaning terms [Sanderson 1994].

5. Summary

Described methods of automated semantic relations acquisition have various performance. In the case of highly effective methods, such as: building a domain dictionary, finding collocations, identifying synonym groups or building definitional sequences, it is possible to gather relatively numerous semantic relations and incorporate them into semantic network or other structure.

Some methods, for instance conditional co-occurrence or key phrases method, perform not so well. They can, however, be still enhanced by improving their mechanisms, especially by taking such omitted effect like homonymy into account. Word sense ambiguity is important reason of IR performance decrease and is subject to further research.

Semantic knowledge, represented in a structure such as semantic network can be utilized to improve the efficiency of information retrieval systems as, for instance, by expanding user queries in order to refine the results, and, as shown in the introduction, the improvement of retrieval recall and precision, positively reflects on the profitability of information systems.

6. References

[Baeza-Yates 1999] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, ACM Press, Addison-Wesley Longman Publishing Co., New York 1999

[Cimiano 2003] P. Cimiano, S. Staab, J. Tane, *Deriving Concept Hierarchies from text by Smooth Formal Concept Analysis*, Karlsruhe 2003

[Cormen 2001] T. H. Cormen, C. E. Leiserson, R. L. Rivest., *Wprowadzenie do algorytmów*, WNT, Warszawa 2001

[Daconta 2003] M. C. Daconta, L. J. Obrst, K. T. Smith, *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*, John Wiley & Sons, 2003

[Evert 2001] S. Evert, B. Krenn, „Methods for the qualitative evaluation of lexical association measures", *39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, 2001, http://www.collocations.de/AM/

[Frakes 1992] W. B. Frakes, R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, 1992

[Hearst 1992] M. A. Hearst, „Automatic Acquisition of Hyponyms from Large Text Corpora", *Proceedings of the Fourteenth International Conference on Computational Linguistics*, Nantes 1992

[Hammerl 1993] R. Hammerl, J. Sambor, *O statystycznych prawach językowych*, Polskie Towarzystwo Semiotyczne, Warszawa 1993

[Hotho 2003] A. Hotho, A. Maedche, S. Staab, „Ontology-based Text Document Clustering", *Proceedings of the Conference on Intelligent Information Systems*, Zakopane, Physica/Springer, 2003

[Jing 1994] Y. Jing, W. B. Croft, *An Association Thesaurus for Information Retrieval*, Amherst 1994

[Koo 2003] S. O. Koo, S. Y. Lim, S. J. Lee, „Building an Ontology based on Hub Words for Information Retrieval", *IEEE/WIC International Conference on Web Intelligence*, Halifax 2003

[Letsche 1996] T. A. Letsche, M. W. Berry, „Large-Scale Information Retrieval with Latent Semantic Indexing", *Information Sciences – Applications*, 1996, http://www.cs.utk.edu/~berry/lsi++/

[Mandala 1999] R. Mandala, T. Toukunaga, H. Tanaka, "Combining Multiple Evidence from Different Types of Thesaurus for Query Expansion", *SIGIR'99*, ACM, Berkley 1999

[Naber 2004] D. Naber, "OpenThesaurus: Building a Thesaurus with a Web Community", *OpenThesaurus*, 2004, http:// www.openthesaurus.de/

[Qiu 1995] Y. Qiu, H. Frei, *Improving the Retrieval Effectiveness by a Similarity Thesaurus*, Technical Report 225, Departament Informatik ETH Zürich, 1995

[Sanderson 1994] Mark Sanderson, "Word Sense Disambiguation and Information Retrieval", *Proceedings of the 17th International ACM SIGIR*, Dublin 1994

[Sanderson 1999] M. Sanderson, B. Croft, „Deriving concept hierarchies from text", *SIGIR'99*, ACM, Berkley 1999

[Schuetze 1995] H. Schütze, J. O. Pedersen, „Information Retrieval Based on Word Senses", *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas 1995

[Weiss 2005] D. Weiss, *Stempelator: A Hybrid Stemmer for the Polish Language*, Technical Report RA-002/05, Politechnika Poznańska, 2005

[Yarowsky 1992] D. Yarowsky, „Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora", *Proceedings of COLING-92*, Nantes 1992

[Zadeh 1965] L. A. Zadeh, „Fuzzy sets", *Information and Control* 8, 1965