A Semantic Method for Textual Entailment

Andrew Neel, Max Garzon, and Vasile Rus

Department of Computer Science
373 Dunn Hall
Memphis, Tennessee 38152-3240
ajneel@acm.org, {mgarzon, vrus}@memphis.edu

Abstract

The problem of recognizing textual entailment (RTE) has been recently addressed using syntactic and lexical models with some success. Here, we further explore this problem, this time using the world knowledge captured in large semantic graphs such as WordNet. We show that semantic graphs made of synsets and selected relationships between them enable fairly simple methods that provide very competitive performance. First, assuming a solution to word sense disambiguation, we report on the performance of these methods in the four basic areas of information retrieval (IR), information extraction (IE), question answering (QA), and multi-document summarization (SUM), as described using benchmark datasets designed to test the entailment problem in the 2006 RTE (Recognizing Textual Entailment) challenge. We then show how the same methods yield a *solution to word sense disambiguation, which* combined with the previous solution yields a *fully automated* solution with about the same performance.

Introduction

The task of recognizing textual entailment (RTE) as defined by (Dagan et al. 2005; Bar-Haim et al. 2006) is the task of determining whether the *meaning* of one *text* (the *hypothesis*) is entailed (or inferred) from another text (simply, the *text*) to humans. One application of entailment is question-answering or automated tutoring systems where the objective is to determine if the answer provided entails the expected or ideal answer (Graesser et al. 2000; Graesser, Hu, and McNamara 2005). A second example is multi-document summarization where it is desirable to remove any sentence from the summary that could be inferred by another sentence. Still another is information retrieval where it is desirable to extract only documents that entail the query.

One of the related challenges of RTE is *Word Sense Disambiguation* (WSD). Solutions to WSD literally require a world of knowledge (or an extensive encyclopedia of world events) to match words and phrases to meanings, herein called *synsets*. WSD also requires a protocol for disambiguating words and phrases into synsets *automatically, i.e.,* without human assistance. Humans implicitly disambiguate words by matching the word *in context* to *meanings* and *experiences* stored in memory. Here, context and experience serve as the world knowledge to humans. Consider the following entailment example: the statement "*John Smith received a belt.*" entails "*John Smith received a strip of leather to fasten around his waist.*" In this example from RTE, "belt" may have the synsets of "a strip of leather to fasten around the waste", "a strip of leather with machine-gun ammo attached" or "a strong punch". The human may remember the full set of *meanings* but his experience will quickly identify "a strip of leather to fasten around the waste" as the specific and proper meaning. Resolving word/phrases to a list of synsets (i.e., a concept or meaning) is relatively easy. However, no automated solution has captured human experience sufficiently deep to choose the correct synset from the list. Therefore, the crux of this issue is finding a representation of human experience in a model that will perform for computers, and with comparable success, the same reasoning function performed by humans.

Humans are very good at solving both entailment *and* WSD because we seem to be able to relate words and lexicon into what is meant by the speaker in the context of prior knowledge of the real world. This paper presents a solution for entailment that can be implemented easily by digital computer systems. Arguably, the closest digital equivalent to a human's experience with word relationships is WordNet, where the fundamental construct is not a word but an abstract semantic concept. Each concept, called synset (synonyms set), may be expressed by different words, and, conversely, the same word may represent different concepts. As the name implies, the concepts of WordNet are inter-connected to provide a network relationships between concepts. In this paper, we use semantic models of world knowledge as exemplified by WordNet to show that semantic graphs made of synsets and selected relationships between them enable fairly simple methods that provide very competitive performance to the problem of word sense disambiguation. Assuming a solution to WSD, we then show how these methods significantly improve the performance of entailment cases in the four basic areas of information retrieval (IR), information extraction (IE), question answering (QA), but mostly in multi-document summarization (SUM), as described using benchmark datasets designed to test the problems in the 2006 RTE challenge.

This paper is organized as follows. First, we present a brief summary of an entailment challenge issued to help researchers in evaluating solutions the entailment problem. As part of this discussion, we describe briefly the effectiveness of conventional solutions to solve the entailment problem. Next, a short summary of WordNet is presented. Finally, we show a simple protocol that can be powerful enough to solve both WSD and entailment with results that are very comparable to conventional solutions.

Background and Related Work

RTE Challenge

In 2005, a first challenge was put forth (Dagan et al. 2005) to researchers to find a method that resolves or approximately resolves entailment. The full problem is obviously hard, even for the average human. In order to be able to make objective comparisons between different solutions, a standard test data was published and has since been updated annually (most recently by Bar-Haim et al., 2006). Each data set consists of a fixed number of tuples (800 in the 2006 set) divided into ontological categories. Each tuple consists of a "text" paragraph (T), one additional paragraph called "the hypothesis" (H), and the judgment about whether T entails H (D). At the time the research of this paper was being performed, a 2007 RTE Challenge had been issued but results were not available. As a result, this research is focused on the results published for 2006 RTE challenge (Bar-Haim et al., 2006). In this paper, the results of over 40 submissions to the challenge are compared with a test data set (available at http://www.pascal-network.org/Challenges/RTE2/). This data set includes 800 tuples divided into four categories, each containing 100 positive and 100 negative examples from one of the following four applications: Information Extraction (IE), Information Retrieval (IR), Question Answering (QA), and Summarization (SUM). We selected a subset of 800 tuples (100 positive and 100 negative tuples for each of the four categories) as a training set. For a description of the characteristics and challenges of each category, the reader is referred to (Bar-Haim et al., 2006)

In addition to the individual challenges of each category, any method faces the additional challenge of performance across all categories. Thus, we consider a fifth category consisting of the full set of 800 tuples.

Conventional solutions

In the 1990s, meaning representations were approached by methods, such as selecting and counting keywords. These solutions would assume some meaning similarity existed between text fragments if *words* in one fragment overlapped a certain percentage of those in the other. More recent solutions have moved beyond mere lexicon and syntax into semantic models that attempt to dynamically resolve word meanings. Among the leaders of these semantic models is Latent Semantic Analysis, or LSA, (Deerwester et al., 1990) which captures the "synonym meanings" of words by the company each word keeps during a training phase. The ability of LSA to capture word meanings requires large quality input in the training phase. While this approach does not explicitly disambiguate words into a specific word meaning, it does improve meaning similarity results by correlating words that express the same concept in different ways.

Textual entailment differs from text similarity in that the relationship between the two fragments under consideration is unidirectional. One text T may entail another H, but H may not necessarily entail T. Recently, (Raina et al. 2005) attacked the problem on two fronts (lexical and semantic). The lexical approach used parse trees to capture syntactic dependencies and sentence structure. This structure was semantically annotated. The result protocol used logical formulas to achieve results that proved the highest confidence weighted score in the RTE Challenge of 2005.

(Rus and Graesser, 2006) have addressed the issue of sentence structure. They proposed capturing sentence structure by mapping student answers and expectations (i.e., ideal answers) in AutoTutor, an intelligent tutoring system, into lexico-syntactic graphs. The degree of relatedness is measured by the degree of graph subsumption. This approach yields good results without an external knowledge source. However, this approach does not disambiguate terms before comparison and has not been demonstrated to address negation, although the authors indicate this might be possible.

By examination of the competitive submissions to the RTE challenge in 2006, we can gauge the approaches considered best by experts in the field. Although not a true scientific survey *per se*, the result is nevertheless telling of the current state of available solutions to entailment problems. Table 1 shows the result of an ontological sort of the more than 40 submissions to the RTE-2 2006 challenge. Results include semantic approaches (similar to LSA described above), lexical and syntactic (Rus and Graesser, 2006), and logical inference models (such as proposed by (Raina et al., 2005)). Most submissions used a combination of techniques. The best scoring approaches used either inference

- **Lexical Relations (32)**
- Subsequence overlap (11)
- **Syntactic Matching (28)**
- Semantic Role Labeling (4)
- Logical Inference (2)
- **Web-Based Statistics (22)**
- **Machine Learning Classification (25)**
- Paraphrase Template (5)
- Acquisition of Entailment (1)

Table 1: More that 40 submissions from over 20 experts are categorized by their approach to the entailment problem. The four most popular are highlighted in bold. Topping the list is "Lexical Relations".

models or included virtually every other model. The very best score of these submissions was accurate to about 0.75. The most frequent approach is "lexical relations", an approach that essentially infers meaning from word-level relationships. The second most frequent approach uses syntax to match syntactic contributions (such as grammar and negation) to infer a semantic relationship if one were to exist. It is clear that the focus of conventional wisdom is to infer meaning from syntactic context and lexical comparisons.

Regardless of the approach, any general purpose solutions to the recognizing textual entailment problem will need to address word-sense disambiguation (assignment of proper meaning to words), sentence structure (context, tense, etc.), and sentence negation. As can be seen with many of the approaches described in Table 1, meaning is inferred without directly addressing the word-sense disambiguation problem. Any ideal solution would need to assume that *a priori* knowledge of the particular topic would not be provided. Literally, the world of knowledge is relevant. The critical issue appears to be disambiguation (assignment of meaning). Once WSD is solved, the equally challenging issues of sentence structure and negation persist. An ideal solution may well require addressing all of the above.

Word Sense Disambiguation (WSD)

A chief problem in assessing entailment through automation is the complexity inherent to language. For example, determining the meaning (or sense) of a word or phrase used in context of a sentence or paragraph. Most words, and phrases, are ambiguous in that they have more than one meaning. The task of WSD is that of determining which *meaning,* with respect to some glossary of meanings such as WordNet, a given word or phrase is intended where more than one interpretation is possible. For example, the word *kill* is ambiguous (without context) and can convey the concepts of "*murder"*, "*incidental death by accident"* or even "*intentional death through legal channels"*. In context of a particular example, "*She killed her husband."* can only convey one of these meanings (*murder)*. Most humans are excellent at WSD. The challenge is for intelligent systems to make that determination automatically.

A first step towards automated WSD is to construct a list of *all* possible meanings and their inter-relationship for *all* terms and for *all* languages. This challenge has been actively pursued at Princeton University under a project called WordNet (http://WordNet.princeton.edu) for over a quarter century.

WordNet is a large lexical database of English developed by George Miller (Beckwith and Miller 1990) to express nouns, verbs, adjectives and adverbs. They are grouped into sets of cognitive synonyms (*synsets*). Each synset expresses a distinct *concept*. Synsets are interlinked by means of lexico-semantic relations. The result can be thought of as a directed graph where the vertices are synsets and the edges/arcs are semantic relationships between the synsets (not necessarily symmetric.)

Word relationships found in WordNet proved helpful in expanding queries for the task of Information Retrieval (Voorhees, 1994). Even without WordNet, WSD has been demonstrated experimentally to improve results for the tasks of Information Retrieval (Schütze and Pederson, 1995; Gonzalo et al. 1998) and Question Answering (Negri, 2004); though some have suggested the opposite that WSD is useless for Information Retrieval and possibly harmful in some cases (Salton, 1968; Salton and McGill, 1983; Krovetz and Croft, 1992; Voorhees, 1993). In (Schütze and Pederson, 1995), precision for the task of standard IR test collection (TREC-1B) improved by more than 4% using WSD. It is thus intriguing what advantages may exist in treating the entailment problem using a purely semantic approach rather than the more common syntactic and lexical approach.

Entailment assuming Word Disambiguation

A natural question about entailment is thus to quantify precisely the benefit that word sense disambiguation (WSD) has to entailment problems. This approach essentially ignores negation, sentence structure and resolves entailment solely on the contribution of WSD. This section addresses this question by presenting a simple *inclusion* procedure to determine if a hypothesis (H) is entailed (or can be inferred) by a Text (T) assuming word disambiguation has already been resolved. This protocol assumes that meanings of terms have been assessed *a priori* by humans. The algorithm determines the percentage of the overlap of synsets in the hypothesis with the synsets in the text.

Table 2 demonstrates the advantages of this approach. Key terms from two short paragraphs are disambiguated by human assessment using synsets provided by WordNet. In this example, *gunman* and *help* convey the same meaning as *shooters* and *aid* (respectively.) However, lexical term matching will not identify the terms as a match. Pure lexical comparison would only match four of the six words in the hypothesis. By using our protocol for WSD, all terms were matched despite very different words.

A more formal description of this entailment solution follows. A bi-partite graph is constructed for each tuple with one part corresponding to the text and the other to the hypothesis. The vertices are *synsets, not* words. Each part has an independent set of vertices where each represents one synset. The semantic relationships that would relate one word to another are represented by edges. Edges only connect vertices across regions. Both the semantic relationship and synsets associated with the Text and Hypothesis are determined by human assessment. Entailment is determined by how connected the synsets in part H are to the synsets in part T. The more H-vertices included in (connected to) T; the more likely it is that T entails H. Entailment is assumed to be false until enough of H connects to T. A threshold for how many connections are required to determine that an entailment is present was optimized ex-

Text: The shooters escaped as other soldiers tried to give aid to the wounded.
Hypothesis: The gunmen escaped as the other soldiers tried to give help to the wounded.

TEXT	SYNSETS FROM WORDNET	HYPOTHESIS	*Lexical*	*Semantic*
shooters	(n) gunman, gunslinger, hired gun, gun, gun for hire, triggerman, hit man, hitman, torpedo, shooter (a professional killer who uses a gun)	gunmen		✓
escaped	(v) escape, get away, break loose (run away from confinement)	escaped	✓	✓
soldiers	(n) soldier (an enlisted man or woman who serves in an army)	soldiers	✓	✓
tried	(v) try, seek, attempt, essay, assay (make an effort or attempt)	tried	✓	✓
give	(v) give (transfer possession of something concrete or abstract to somebody)			
aid	(n) aid, assist, assistance, help (The activity of contributing to the fulfillment of a need or furtherance of an effort or purpose)	help		✓
wounded	(n) wounded, maimed (people who are wounded)	Wounded	✓	✓

Table 2: Two simple paragraphs (Text and Hypothesis) demonstrate the advantages of disambiguation to entailment. The words (first and third column) of the paragraphs are aligned in rows according to their semantic meaning (second

perimentally. This threshold is optimized with a training dataset, mentioned above, so that the solution would return the maximum number of correct answers. Later, the thresholds obtained from this optimization step (using the training dataset (the 100 positive and 100 negative examples described above) were used to assess entailment with the full test set (Table 3).

Table 3 shows the result of this experiment for each category. This procedure answers correctly 3 times more frequently than incorrect. This procedure appears to work best for Information Retrieval (IR) and Multi-document summarization (SUM).

	IE	IR	QA	SUM	Overall
(A) Total Correct	45%	70%	67%	80%	72.90%

Table 3: Entailment by a simple semantic procedure is significantly better than other methods on the RTE challenge 2006 dataset, assuming a solution to disambiguation.

The inclusion procedure is a simple but clear demonstration that the human ability for assessing entailment lies in the human's capability to disambiguate words. This inclusion algorithm further shows that entailment could be assessed *automatically* by substituting the human assessment described in this section by a similar automatic WSD procedure. Though the next section will focus on one such word disambiguation solution, *any* word disambiguation method capable of assessing semantic meaning with limited context could be coupled with this inclusion procedure to result in a fully automated entailment solution.

Automated Word Disambiguation

The easiest algorithm for WSD is to assume every word in the tuple bears the most popular meaning according to the language. The first sense in WordNet for each word represents the most popular word meaning. This first algorithm was tested and shown not to help significantly. The general problem was that this approach did not capture enough of the context to discern when less frequent word meanings were used. This section presents a refinement of this procedure for disambiguating words automatically by combining *the context* of the Text and Hypothesis contained in the tuple with the word meanings and *relationships* captured in WordNet.

Automated WSD with WordNet Synsets

This procedure for WSD first extracts the words from the 800 tuples. Phrases were identified that express a single meaning with multiple words (e.g. "*au jus" or "hard disk").* Articles, prepositions, and other common terms were sanitized from the word set. Hyphenated words were broken into their components. In several cases, the input set contained spelling errors. These errors were preserved. The final word set contained about 5,500 unique words and phrases.

We then used a simple script written in PERL to extract from WordNet all synsets for each word. Each synset was stored in hash table and indexed by the word or phrase used to query WordNet. Each word that did not have any synset defined in WordNet was ignored and removed from the word set.

Word disambiguation is performed by the following protocol. The set of synsets associated with each word in the Hypothesis are compared with each set of synsets associated with the words in the Text. The resulting overlap (intersection) of synsets represents the meanings those words have in common. The most popular synset or meaning (according to WordNet) *in the context of the paragraphs* is selected from the intersection. Thus, synsets that are more popular *according to WordNet* are probably ignored when others are more popular *according to the context of the paragraphs*. The selected synset is assumed to be the intended meaning for words found in the Text and Hypothesis. This procedure was performed for each set of synsets in the Hypothesis until all words have been disambiguated or the possibility for disambiguation has been exhausted after testing the full set of synets retrieved from WordNet. Words that exhaust all opportunities but remain ambiguous were assigned the most popular meaning in WordNet. The result of this procedure is an "optimistic" disambiguation based on the context provided in the tuple.

Semi-Automated WSD with WordNet Synsets

Here, the results of the previous procedure were examined manually to determine any additional semantic relationships that could result in better disambiguation. The automated method for word disambiguation described above was performed first. Afterwards, human assessors reviewed the result by examining the subset of words whose resulting meaning was not correctly selected. Words that are clearly related but did not share a common synset were identified and paired together. The result of this procedure is an optimistic disambiguation based on the context provided in the tuple with additional experience from humans.

WordNet was then searched for semantic relationships (such as hypernymy, synonym, antonym, etc) that would express the semantic relationship of the pair. In less than 25% of the cases, the relationship was expressed through hypernymy relationships. The majority of cases required information that was simply not available in WordNet. This fact highlights a potential limitation on WordNet's ability to capture and provide semantic meaning and relationships.

Automated WSD with Semantic Relationships

This procedure automates word disambiguation with hypernymy relationships captured in WordNet. Essentially, the automated WSD performed above is modified so that synsets related by hypernymy relationships are included. Hypernymy relationships express abstraction and generalization. Each relationship was exhaustively explored to extract related synsets. Hypernyms were stored in separate hash and indexed by the source word. Word disambiguation is performed using the same procedure as before but including hypernyms with the synsets. The result of this procedure is an optimistic disambiguation based on the context provided in the tuple and additional human experience captured in WordNet.

Entailment with Automated WSD

So far, entailment was determined by the degree of connectedness in a bi-partite graph whose vertices are disambiguated words and whose edges express "sameness" in meaning. In that instance, disambiguation was preprocessing performed by humans. Here, we replace the human judgment with automated WSD procedures and compare the performance of the resulting procedures for entailment. In this test, we used the full set of 800 tuples. Here, edges express only the "equivalent" relationships between synsets (e.g. "kill" connects to "murder" only if both disambiguate to the same synset). The thresholds for determining whether entailment exists in this step had been assessed in the training phase and described in the section "Entailment assuming word disambiguation" above.

Table 4 row B (4.B) shows the performance of entailment using this procedure. Three of the four categories of data determined entailment better than just random guessing which is expected to be correct about 50% of the time. Information retrieval (IR) and summarization (SUM) were more than 15% above the simple guessing. Overall scores show a near 10% improvement over guessing. Only one case (IE) performed worse than guessing (> 5%). When compared to the same protocol using human disambiguation (Table 4.A), the scores for Information Extraction (IE) *improve by* 2%. The remaining three categories performed worse by 5-10% when compared with the protocol that assumes human disambiguation. Overall scores are about 10% worse.

When this procedure was replaced with the semi-automated procedure (shown in Table 4.C), scores for Information Retrieval and Summarization improved while Question Answering and Information Extraction remained the same. When fully automated with hypernyms (Table 4.D), scores for Information Retrieval (IR) and Question Answering (QA) declined from the semi-automated procedure by 1%. Summarization (SUM) declines by 4% while Information Extraction (IE) remains in stasis. Overall scores remained better than simple guessing, but there is room for improvement to meet performance levels where disambiguation is performed by humans.

This procedure is clearly competitive with solutions submitted to the RTE challenge. Its simple approach outperformed 28 of the 41 submissions to the 2006 RTE Challenge. In fact, the next 10 submissions were only better by < 2% than our overall score.

	IE	IR	QA	SUM	Overall
(A) Total Correct	45%	70%	67%	80%	72.90%
(B) Total Correct	47%	65%	56%	70%	59.10%
(C) Total Correct	47%	67%	56%	72%	60.10%
(D) Total Correct	47%	66%	55%	68%	58.80%

Table 4: Scores are shown for three procedures B, C, and D in determining entailment automatically. These scores compare favorably to the scores of entailment assuming disambiguation (A) and are competitive with other solutions submitted to the 2006 RTE Challenge.

Analysis and Conclusions

In an attempt to determine the cause of the gap (about 14%) between entailment assuming disambiguation (4.A) and entailment with WordNet (4.D), the automatic WSD procedure above was expanded to include human assessment. As previously discussed, the automatic WSD approach relies on WordNet to provide information about the *use and relationships among* words in the English language (specifically a word's specific meaning in context of a sentence or paragraph). As a result, the short-comings of either inputs from WordNet or inputs from the text will influence the result. Our analysis identified two causes:

(1) Words or phrases that were used incorrectly

(2) Words or phrases that were used correctly but were not properly defined or semantically tied to related concepts in WordNet.

An example of #2 is easily demonstrated with two words: *toxic waste* and *pollution.* The two words could not be semantically linked through WordNet though clearly related semantically. In other cases, *environment* and *reality* were used synonymously (though not related when defined with main stream dictionaries). It is clear that improving the content in WordNet would have a direct positive impact on the results presented in this paper. How much improvement can be expected remains an open question.

The authors feel that such information may still be automatically obtained from WordNet as it is today using custom tools to retrieve data and relationships not otherwise available. It may be useful to explore other semantic relationship using methods similar to procedure D (which incorporated hypernyms). One unfortunate barrier to the immediate testing of these other semantic links is the sheer size of WordNet. Additional research may be needed to fully explore these problems.

Conclusions

We addressed the recognizing textual entailment problem using WordNet as a semantic model of world knowledge. We show that semantic graphs made of synsets and selected relationships between them enable fairly simple methods to assess entailment competitively. We have presented three very simple models for word disambiguation. Lastly, we produced a fully automated solution to entailment that produces results competitive with the best known lexical and syntactic based models. The results of this work constitute a further confirmation that WSD is powerful asset to Natural Language processing tasks. The success of this simple approach is encouraging and may yet lead to more complex algorithms which may result in even better performance and possibly with a broader set of textual entailment problems.

References

1. Bar-Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B. & Szpektor, I. (2006). The Second PASCAL Recognizing Textual Entailment Challenge. *In Proceedings of the Second PASCAL Challenges Workshop on Recognizing Textual Entailment*.
2. Beckwith, R. & Miller, G.A. (1990). Implementing a lexical network. *International Journal of Lexicography* 3 (4), 302-312.
3. Dagan, I., Glickman. O., & Magnini, B. The PASCAL Recognizing Textual Entailment Challenge. *In Proceedings of the PASCAL Challenges Workshop on Recognizing Textual Entailment*.
4. Deerwester, S., Dumais, S., Furnas, G.W., Landauer, T.K., & Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the Society for Information Science* 41 (6), 391-407.
5. Graesser, A., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D.,Person, N., and the Tutoring Research Group. (2000). Using Latent Semantic Analysis to evaluate the contributions of students in AutoTutor. Interactive Learning Environments, 8 ,149-169.
6. Gonzalo, J., Verdejo, F., Chugur, I., & Cigarran, J. (1998). Indexing with WordNet synsets can improve text retrieval. *In Proceedings of COLING-ACL '98 Workshop on Usage of Word.Net in Natural Language Processing Systems,* Montreal, Canada, August.
7. Krovetz, R. and Croft, W. B. 1992. Lexical ambiguity and information retrieval. ACM Transactions on Information Systems 10, 2, 115-141.
8. Negri, M. (2004). Sense-based blind relevance feedback for question answering. In SIGIR-2004 *Workshop on Information Retrieval For Question Answering* (IR4QA), Sheffield, UK.
9. Raina, R., Ng, A., & Manning, C.D. (2005). Robust textual inference via learning and abductive reasoning. *Proceedings of the Twentieth National Conference on Artificial Intelligence*. AAAI Press.
10. Rus, V., Graesser, A., & Desai, K. (2005). Lexico-Syntactic Subsumption for Textual Entailment. *Proceedings of Recent Advances in Natural Language Processing (RANLP 2005)*, Borovets, Bulgaria.
11. Rus, V. & Graesser, A. (2006). Deeper Natural Language Processing for Evaluating Student Answers in Intelligent Tutoring Systems. *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, Boston, Massachusetts, USA.
12. Salton, G. 1968. Automatic Information Organization and Retrieval. McGraw-Hill, New York.
13. Salton, G. and McGill, M. 1983. Introduction to Modern Information Retrieval. McGraw-Hill, New York.
14. Schütze, H. & Pederson, J. (1995). *Information Retrieval based on word senses*. Symposium on Document Analysis and Information Retrieval (SDAIR). Las Vegas, Nevada, 161-175.
15. Voorhees, E. (1994). Query expansion using lexical-semantic elations. In *Proceedings of the 17th SIGIR Conference*, Dublin, Ireland, (June, 1994).
16. Voorhees, E. M. 1993. Using WordNet to disambiguate word senses for text retrieval. In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, 171-180.