Using WordNet for Text Categorization

Zakaria Elberrichi[1], Abdelattif Rahmoun[2], and Mohamed Amine Bentaalah[1]
[1]EEDIS Laboratory, Department of Computer Science, University Djilali Liabès, Algeria
[2]King Faisal University, Saudi Arabia

Abstract**:** *This paper explores a method that use WordNet concept to categorize text documents. The bag of words representation used for text representation is unsatisfactory as it ignores possible relations between terms. The proposed method extracts generic concepts from WordNet for all the terms in the text then combines them with the terms in different ways to form a new representative vector. The effects of this method are examined in several experiments using the multivariate chi-square to reduce the dimensionality, the cosine distance and two benchmark corpus the reuters-21578 newswire articles and the 20 newsgroups data for evaluation. The proposed method is especially effective in raising the macro-averaged F1 value, which increased to 0.714 for the Reuters from 0.649 and to 0.719 for the 20 newsgroups from 0.667.*

Keywords**:** *20Newsgroups, ontology, reuters-21578, text categorization, wordNet, and cosine distance.*

Received April 5, 2006; Accepted August 1, 2006

1. Introduction

Text Categorization (TC) is the classification of documents with respect to a set of one or more pre-existing categories [14]. TC is a hard and very useful operation frequently applied to assign subject categories to documents, to route and filter texts, or as a part of natural language processing systems.

During the last decades, a large number of methods proposed for text categorization were typically based on the classical *Bag-of-Words* model where each term or term stem is an independent feature. The disadvantages of this classical representation are:

- The ignorance of any relation between words, thus learning algorithms are restricted to detect patterns in the used terminology only, while conceptual patterns remain ignored.
- The big dimensionality of the representation space.

In this article, we propose a new method for text categorization, which is based on:

- The use of the WordNet ontology to capture the relations between the words.
- The use of the multivariate χ_2 method to reduce the dimensionality and create the categories profiles.

The originality of this approach lies in merging terms with their associated concepts extracted from the used ontology to form a hybrid model for text representation.

In order to show the positive contribution of this approach, we have performed a series of experiments on the Reuters-21578 and 20Newsgoups test collections. WordNet's large coverage and frequent utilization has led us to use it for our experiments.

The remainder of this paper is structured as follows. Section 2 presents a brief presentation of WordNet. The architecture of our approach is provided in section 3 with its different stages. Testing and performance analysis compared to the *Bag-Of-Word* representation is provided in section 4. Section 5 cites some related works. The conclusion and future work are provided in section 6.

2. WordNet

There exist many difficulties to surmount to create an effective texts categorization system: the speed of the indexing and research, the index size, the robustness, the reliability, the effectiveness,…etc. But the principal difficulties encountered in the field are those posed by the natural languages themselves. This is why many experiments using linguistic resources and treatments were realized and presented in the literature. The use of knowledge and advanced linguistic treatments in the field does not achieve the unanimity in the community. Indeed, many experiments seem to show that sometimes the results obtained instead of improving do degrade. This was not the case of our approach, where we used two of the semantic relations of WordNet: the synonymy and the hyponymy.

WordNet is a thesaurus for the English language based on psycholinguistics studies and developed at the University of Princeton [11]. It was conceived as a data-processing resource which covers lexico-semantic categories called *synsets*. The *synsets* are sets of synonyms which gather lexical items having similar significances, for example the words *"a board"* and *"a plank"* grouped in the *synset {board, plank}*. But *"a board"* can also indicate a group of people (e.g., a

board of directors) and to disambiguate these homonymic significances *"a board"* will also belong to the *synset {board, committee}*. The definition of the *synsets* varies from the very specific one to the very general. The most specific *synsets* gather a restricted number of lexical significances whereas the most general *synsets* cover a very broad number of significances.

The organization of WordNet through lexical significances instead of using lexemes makes it different from the traditional dictionaries and thesaurus [11]. The other difference which has WordNet compared to the traditional dictionaries is the separation of the data into four data bases associated with the categories of verbs, nouns, adjectives and adverbs. This choice of organization is justified by psycholinguistics research on the association of words to the syntactic categories by humans. Each database is differently organized than the others. The names are organized in hierarchy, the verbs by relations, the adjectives and the adverbs by *N*-dimension hyperspaces [11].

The following list enumerates the semantic relations available in WordNet. These relations relate to concepts, but the examples which we give are based on words.

- *Synonymy*: relation binding two equivalent or close concepts (frail /fragile). It is a symmetrical relation.
- *Antonymy*: relation binding two opposite concepts (small /large). This relation is symmetrical.
- *Hyperonymy*: relation binding a concept$_{-1}$ to a more general concept$_{-2}$ (tulip /flower).
- *Hyponymy*: relation binding a concept$_{-1}$ to a more specific concept$_{-2}$. It is the reciprocal of hyperonymy. This relation may be useful in information retrieval. Indeed, if all the texts treating of vehicles are sought, it can be interesting to find those which speak about cars or motor bikes.
- *Meronymy*: relation binding a concept$_{-1}$ to a concept$_{-2}$ which is one of its parts (flower/petal), one of its members (forest /tree) or a substance made of (pane/glass).
- *Metonymy*: relation binding a concept$_{-1}$ to a concept$_{-2}$ of which it is one of the parts. It is the opposite of the meronymy relation.
- *Implication*: relation binding a concept$_{-1}$ to a concept$_{-2}$ which results from it (to walk /take a step).
- *Causality*: relation binding a concept$_{-1}$ to its purpose (to kill /to die).
- *Value*: relation binding a concept$_{-1}$ (adjective) which is a possible state for a concept$_{-2}$ (poor /financial condition).
- *Has the value*: relation binding a concept$_{-1}$ to its possible values (adjectives) (size /large). It is the opposite of relation value.
- *See also*: relation between concepts having a certain affinity (cold /frozen).
- *Similar to*: certain adjectival concepts which meaning is close are gathered. A *synset* is then designated as being central to the regrouping. The relation '*Similar to*' binds a peripheral *synset* with the central *synset* (moist /wet).
- *Derived from*: indicate a morphological derivation between the target concept (adjective) and the concept origin (coldly /cold).

2.1. Synonymy in WordNet

A *synonym* is a word which we can substitute to another without important change of meaning. Cruse [2] distinguishes three types of synonymy:

- Absolute synonymes.
- Cognitive synonymes.
- Plesionymes.

According to the definition of Cruse [3] of the cognitive synonyms, *X* and *Y* are cognitive synonyms if they have the same syntactic function and that all grammatical declaratory sentences containing *X* have the same conditions of truth as another identical sentence where *X* is replaced by *Y*.

Example: *Convey /automobile*

The relation of *synonymy* is at the base of the structure of WordNet. The lexemes are gathered in sets of synonyms ("synsets"). There are thus in a *synset* all the terms used to indicate the concept.

The definition of *synonymy* used in WordNet [11] is as follows: *"Two expressions are synonymous in a linguistic context C if the substitution of for the other out of C does not modify the value of truth of the sentence in which substitution is made*".
Example of synset: [Person, individual, someone, somebody, mortal, human, drunk person].

2.2. Hyponyms /Hyperonyms in Word Net

X is a hyponym of *Y* (and *Y* is a hyperonym of *X*) if:

- F(*X*) is the minimal indefinite expression compatible with sentence *A* is F(*X*) and
- *A* is F(*X*) implies *A* is F(*Y*).

In other words, the hyponymy is the relation between a narrower term and a generic term expressed by the expression "is-a".

Example:
It is a dog → It is an animal [2].

A dog is a hyponym of animal and animal is a hyperonym of dog.

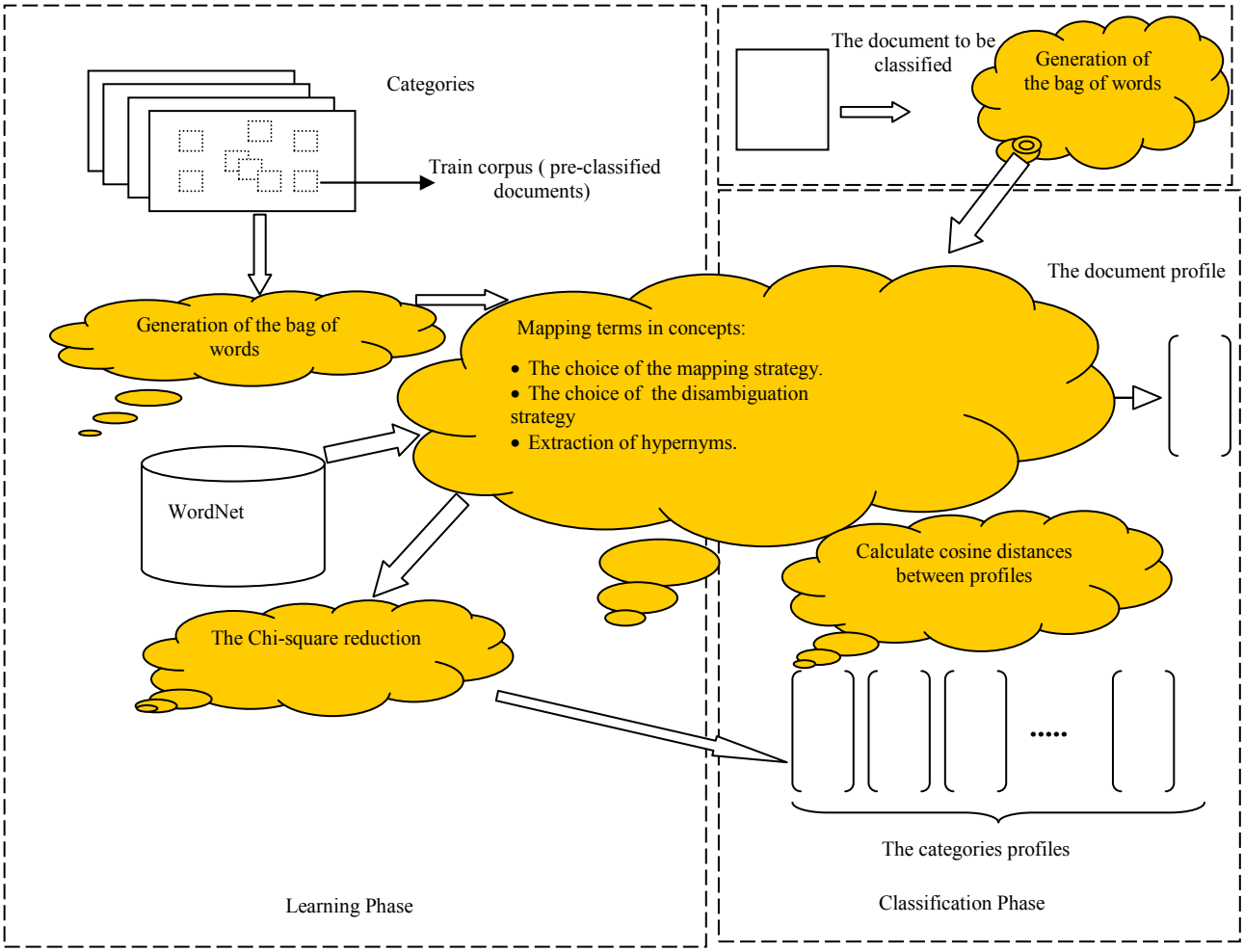

Figure1. The suggested approach.

In WordNet, the hyponymy is a lexical relation between meanings of words and more precisely between synsets (Synonym Sets). This relation is defined by: X is a hyponym of Y if "X is a kind of Y" is true. It is a transitive and asymmetrical relation, which generates a downward hierarchy of heritage for the organization of the nouns and the verbs. The hyponymy is represented in WordNet by the symbol '@', which is interpreted by "is-a" or "is a kind of".

Example:
It is a tree → It is a plant.

3. WordNet-Based Texts Categorization

The approach suggested is composed of two stages, as indicated in Figure 1. The first stage relates to the learning phase. It consists of:

- Generating a new text representation based on merging terms with their associated concept.
- Selecting the characteristic features for creating the categories profiles.

The second stage relates to the classification phase. It consists on:

- Weighting the features in the categories profiles.
- Calculating the distance between the categories profiles and the profile of the document to be classified.

3.1. The Learning Phase

The first issue that needs to be addressed in text categorization is how to represent texts so as to facilitate machine manipulation but also to retain as much information as needed. The commonly used text representation is the *Bag-Of-Words*, which simply uses a set of words and the number of occurrences of the words to represent documents and categories [12]. Many efforts have been made to improve this simple and limited text representation. For example, [6] uses phrases or word sequences to replace single words. In our approach, we use a method that merges terms with their associated concepts to represent texts. To generate a text representation using this method, four steps are required:

- Mapping terms into concepts and choosing a merging strategy.

- Applying a strategy for word senses disambiguation.
- Applying a strategy for considering hypernyms.
- Applying a strategy for features selection.

3.1.1. Mapping Terms into Concepts

The process of mapping terms into concepts is illustrated with an example shown in Figure 2. For simplicity, suppose there is a text consisting in only 10 words: *government* (2), *politics* (1), *economy* (1), *natural philosophy* (2), *life science* (1), *math* (1), *political economy* (1), and *science* (1), where the number indicated is the number of occurrences.

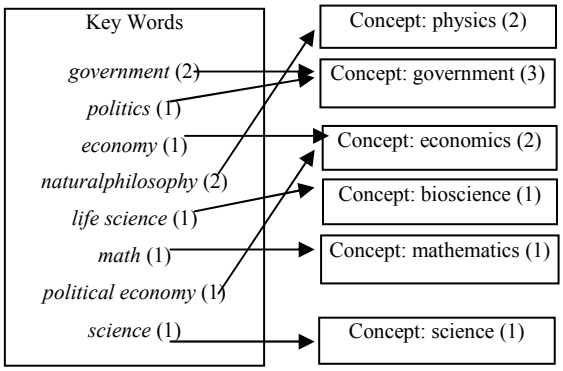

Figure 2. Example of mapping terms into concepts.

The words are then mapped into their corresponding concepts in the ontology. In the example, the two words *government* (2) and *politics* (1) are mapped in the concept *government* and the term frequencies of these two words are added to the concept frequency.
From this point, three strategies for adding or replacing terms by concepts can be distinguished as proposed by [1]:

A. *Add Concept*
This strategy extends each term vector $\vec{t}_d$ by new entries for WordNet concepts *C* appearing in the texts set. Thus, the vector $\vec{t}_d$ will be replaced by the concatenation of $\vec{t}_d$ and $\vec{c}_d$ where $\vec{c}_d = (cf(d, c_1), ..., cf(d, c_l))$. The concept vector with $l = |C|$ and $cf(d, c)$ denotes the frequency that a concept $c \in C$ appears in a text *d*.

The terms, which appear in WordNet as a concept, will be accounted for at least twice in the new vector representation; once in the old term vector $\vec{t}_d$ and at least once in the concept vector $\vec{c}_d$.

B. *Replace Terms by Concepts*
This strategy is similar to the first strategy; the only difference lies in the fact that it avoids the duplication of the terms in the new representation; i.e., the terms which appear in WordNet will be taken into account only in the concept vector. The vector of the terms will thus contain only the terms, which do not appear in WordNet.

C. *Concept Vector Only*
This strategy differs from the second strategy by the fact that it excludes all the terms from the new representation including the terms, which do not appear in WordNet; $\vec{c}_d$ is used to represent the category.

3.1.2. Strategies for Disambiguation

The assignment of terms to concepts is ambiguous. Therefore, one word may have several meanings and thus one word may be mapped into several concepts. In this case, we need to determine which meaning is being used, which is the problem of sense disambiguation [8]. Since a sophisticated solution for sense disambiguation is often impractical [1], we have considered the two simple disambiguation strategies used in [7].

A. *All Concepts*
This strategy considers all proposed concepts as the most appropriate one for augmenting the text representation. This strategy is based on the assumption that texts contain central themes that in our cases will be indicated by certain concepts having height weights. In this case, the concept frequencies are calculated as follows:

$$cf(d, c) = tf\left\{d, \left\{t \in T \mid c \in ref_c(t)\right\}\right\} \quad (1)$$

B. *First Concept*
This strategy considers only the most often used sense of the word as the most appropriate concept. This strategy is based on the assumption that the used ontology returns an ordered list of concepts in which more common meanings are listed before less common ones [10].

$$cf(d, c) = tf\left\{d, \left\{t \in T \mid first(ref_c(t)) = c\right\}\right\} \quad (2)$$

3.1.3. Adding Hypernyms

If concepts are used to represent texts, the relations between concepts play a key role in capturing the ideas in these texts. Recent research shows that simply changing the terms to concepts without considering the relations does not have a significant improvement and some time even perform worse than terms [1]. For this purpose, we have considered the hypernym relation between concepts by adding to the concept frequency of each concept in a text the frequencies that their hyponyms appears. Then the frequencies of the concept vector part are updated in the following way:

$$cf'(d, c) = \sum_{b \in H(c)} cf(d, b) \quad (3)$$

where H(c) gives for a given concept *c* its hyponyms.

3.1.4. Features Selection

Selection techniques for dimensionality reduction take as input a set of features and output a subset of these features, which are relevant for discriminating among categories [3]. Controlling the dimensionality of the vector space is essential for two reasons. The complexity of many learning algorithms depends crucially not only on the number of training examples but also on the number of features. Thus, reducing the number of index terms may be necessary to make these algorithms tractable. Also, although more features can be assumed to carry more information and should, thus, lead to more accurate classifiers, a larger number of features with possibly many of them being irrelevant may actually hinder a learning algorithm constructing a classifier.

For our approach, a feature selection technique is necessary in order to reduce the big dimensionality caused by considering concepts in the new text representation. For this purpose we used the Chi-Square Statistic for feature selection.

The χ_2 statistic measures the degree of association between a term and the category. Its application is based on the assumption that a term whose frequency strongly depends on the category in which it occurs will be useful for discriminating among the categories. For the purpose of dimensionality reduction, terms with small χ_2 values are discarded.

The χ_2 multivariate, noted $\chi^2_{\text{multvariate}}$ is a supervised method allowing the selection of terms by taking into account not only their frequencies in each category but also the interaction of the terms between them and the interactions between the terms and the categories. The principle consists in extracting *K* better features characterizing best the category compared to the others, this for each category.

With this intention, the matrix (term-categories) representing the total number of occurrences of the *p* features in the *m* categories is calculated (see Figure 3). The total sum of the occurrences is noted *N*. The values N_{jk} represent the frequency of the feature X^J in the category e_k. Then, the contributions of these features in discriminating categories are calculated as indicated in Equation 4, then sorted by descending order for each category. The evaluation of the sign in the Equation 4 makes it possible to determine the direction of the contribution of the feature in discriminating the category. A positive value indicates that it is the presence of the feature which contribute in the discrimination while a negative value reveals that it is its absence which contribute in it.

	e_1		e_k		e_m	
X^1	N_{11}	…	N_{1k}	…	N_{1m}	$N_{1.}$
⋮	⋮		⋮		⋮	
X^j	N_{j1}	…	N_{jk}	…	N_{jm}	$N_{j.}$
⋮	⋮		⋮		⋮	
X^n	N_{n1}	…	N_{nk}	…	N_{nm}	$N_{n.}$
	$N_{.1}$		$N_{.k}$		$N_{.m}$	$N = N_{..}$

Figure 3. Matrix of features frequencies in categories.

The principal characteristics of this method are:

- It is supervised because it is based on the information brought by the categories.
- It is a multivariate methode because it evaluates the role of the feature with considering the other features.
- It considers interactions between features and categories.
- In spite of its sophistication, it remains of linear complexity in terms number.

$$C_{jk}^{x^2} = N\frac{(f_{jk} - f_{j.}f_{.k})^2}{f_{j.}f_{.k}} \times sign(f_{jk} - f_{j.}f_{.k}) \qquad (4)$$

where $f_{jk} = \frac{N_{jk}}{N}$ representing the relative frequencies of the occurrences.

3.2. Classification Phase

The classification phase consists in generating a weighted vector for all categories, then using a similarity measure to find the closest category.

3.2.1. Vector Generation

Given the features frequencies in all categories, the task of the *vector generation* step is to create a weighted vector $d = (w(d, t_1), ..., w(d, t_m))$ for any category *d* based on its feature frequency vector $d_{tf} = (tf_d(t_1), ..., tf_d(t_m))$, which commonly results from the feature selection step. Each weight $w(d,t)$ expresses the importance of feature *t* in category *d* with respect to its frequency in all training documents. The objective of using a feature weight rather than plain frequencies is to enhance classification effectiveness.

In our experiments, we used the standard *tfidf* function, defined as:

$$tfidf\ (t_k, c_i) = tf\ (t_k, c_i) \times Log\left(\frac{|C|}{df\ (t_k)}\right) \qquad (5)$$

where:

- $tf(t_k, c_i)$ denotes the number of times feature t_k *occurs in category* c_i.
- $df(t_k)$ denotes the number of categories in which feature t_k occurs.
- $|C|$ denotes the number of categories.

3.2.2. Distance Calculation

The similarity measure is used to determine the degree of resemblance between two vectors. To achieve reasonable classification results, a similarity measure should generally respond with larger values to documents that belong to the same class and with smaller values otherwise.

The dominant similarity measure in information retrieval and text classification is the *cosine similarity* between two vectors. Geometrically, the cosine similarity evaluates the cosine of the angle between two vectors d_1 and d_2 and is, thus, based on angular distance. This allows us to abstract from varying vector length. The cosine similarity can be calculated as the normalized product:

$$S_{i,j} = \frac{\sum_{w \in i \cap j} TFIDF_{w,j} \times TFIDF_{w,j}}{\sqrt{\sum_{w \in i} TFIDF_{w,i}^2} \times \sqrt{\sum_{w \in j} TFIDF_{w,j}^2}} \quad (6)$$

where:
w is a feature, I and J are the two vectors (profiles) to be compared. $TFIDF_{w,i}$ the weight of the term w in I and $TFIDF_{w,j}$ is the weight of the term w in J. This can be translated in the following way:
"*More there are common features and more these features have strong weightings, more the similarity will be close to 1, and vice versa* ".

In our approach, this similarity measure is used to calculate the distance between the vector of the document to be categorized and all categories vector. As a result, the document will be assigned to the category whose vector is the closest with the document vector.

4. Experiments and Evaluation

We have conducted our experiments on two commonly used corpora in text categorization research: 20 Newsgroups, and ModApte version of the Reuters-21578 collection of the news stories. All documents for training and testing involve a pre-processing step, which includes the task of stopwords removal.
Experimental results reported in this section are based on the so-called "F_1 *measure*", which is the harmonic mean of precision and recall.

$$F_1(recall, precision) = \frac{2 \times recall \times precision}{recall + precision} \quad (7)$$

In the above formula, precision and recall are two standard measures widely used in text categorization literature to evaluate the algorithm's effectiveness on a given category where

$$precision = \frac{true\ positive}{(true\ positive) + (false\ positive)} \times 100 \quad (8)$$

$$recall = \frac{true\ positive}{(true\ positive) + (false\ negative)} \times 100 \quad (9)$$

We also use the macroaveraged F_1 to evaluate the overall performance of our approach on given datasets. The macroaveraged F_1 compute the F_1 values for each category and then takes the average over the per-category F_1 scores. Given a training dataset with m categories, assuming the F_1 value for the i-th category is $F_1(i)$, the macroaveraged F_1 is defined as :

$$macroaveraged\ F_1 = \frac{\sum_{i=1}^{m} F_1(i)}{m} \quad (10)$$

4.1. Datasets for Evaluation

4.1.1. Reuters-21578

The Reuters dataset has been used in many text categorization experiments; the data was collected by the Carnegie group from the Reuters newswires in 1987. There are now at least five versions of the Reuters datasets widely used in TC community. We choose the Modapte version of the Reuters-21578 collection of new stories downloaded from http://www.daviddlewis.com/ressources/testcollections/reuters21578. In our experiments, we used the ten most frequent categories from this corpus as our dataset for training and testing as indicated in Table 1.

Table 1. Detailes of the reuters21-578 used categories.

Category	# Training	# Test	Total
Earn	2877	1087	3864
Acquisition	1650	719	2369
Money-fx	538	179	717
Grain	433	149	582
Crude	389	189	578
Trade	369	118	487
Interest	347	131	478
Wheat	212	71	283
Ship	197	89	286
Corn	182	56	238

4.1.2. 20Newsgroups

The 20Newsgroups contains approximately 20,000 newsgroups documents being partitioned (nearly) evenly across 20 different newsgroups, we used the 20newsgroups version downloaded from http://www.ai.mit.edu/~jrennie/20Newsgroups. Table 2 specifies the 20Newsgroups categories and their sizes.

Table 2. Detailes of 20Newsgroups categories.

Category	# Train Docs	# Test Docs	Total # Docs
alt.atheism	480	319	799
comp.graphics	584	389	973
comp.os.ms-windows.misc	572	394	966
comp.sys.ibm.pc.hardware	590	392	982
comp.sys.mac.hardware	578	385	963
comp.windows.x	593	392	985
misc.forsale	585	390	975
rec.autos	594	395	989
rec.motorcycles	598	398	996
rec.sport.baseball	597	397	994
rec.sport.hockey	600	399	999
sci.crypt	595	396	991
sci.electronics	591	393	984
sci.med	594	396	990
sci.space	593	394	987
soc.religion.christian	598	398	996
talk.politics.guns	545	364	909
talk.politics.mideast	564	376	940
talk.politics.misc	465	310	775
talk.religion.misc	377	251	628
Total	**11293**	**7528**	**18821**

4.2. Results

Tables 3 and 4 summarize the results of our approach compared with the *Bag-Of-Word* representation over Reuters-21578 (10 largest categories) and 20Newsgroups categories. The results obtained in the experiments suggest that the integration of conceptual features improved text classification results. On the Reuters categories (see Table 3); the best overall value is achieved by the following combination of strategies: "add concept" strategy using "First concept" strategy for disambiguation with the profile size *k=200*.

Macro-averaged values then reached 71.7%, thus yielding a relative improvement of 6.8% compared to the Bag-Of-Word representation.

The same remarks can be done on the 20Newsgroups categories (see Table 4). The best performance is obtained with the profile size *k=500*. The relative improvement is about 5.2% compared to the Bag-Of-Word representation.

5. Related Work

The importance of WordNet as a source of conceptual information for all kinds of linguistic processing has been recognized with many different experiences and specialized workshops.

There are a number of interesting uses of WordNet in information retrieval and supervised learning. Green [4, 5] uses WordNet to construct chains of related synsets (that he calls 'lexical chains') from the occurrence of terms in a document. It produces a WordNet based document representation using a word sense disambiguation strategy and term weighting. Dave [13] has explored WordNet using synsets as features for document representation and subsequent clustering.

He did not perform word sense disambiguation and only found that WordNet synsets decreased clustering performance in all his experiments. Voorhees [15] as well as Moldovan and Mihalcea have explored the possibility to use WordNet for retrieving documents by keyword search.

It has already become clear by their work that particular care must be taken in order to improve precision and recall.

Table 3.The comparison of performance (F_1) on Reuters-21578.

Term/Concept		Add Concept		Replace Terms By Concepts		Concept Vector Only		Bag-Of-Word
Disambiguation		First	All	First	All	First	All	
The Size of Categories Profiles	K=100	0.703	0.671	0.682	0.658	0.618	0.580	0.643
	K=200	0.709	0.682	0.688	0.670	0.625	0.610	0.659
	K=300	0.717	0.699	0.701	0.690	0.638	0.632	0.665
	K=400	0.718	0.702	0.703	0.694	0.640	0.638	0.666
	K=500	0.719	0.707	0.705	0.698	0.643	0.643	0.666
	K=600	0.719	0.708	0.706	0.699	0.643	0.644	0.667
	K=700	0.719	0.708	0.706	0.699	0.643	0.645	0.667
	K=800	0.719	0.709	0.706	0.699	0.643	0.645	0.667

Table4. The comparison of performance (F_1) on 20Newsgroups.

Term/Concept		Add Concept		Concept Vector Only		Replace Terms By Concepts		Bag-Of-Word
Disambiguation		First	All	First	All	First	All	
The Size of Categories Profiles	K=100	0.714	0.677	0.681	0.708	0.664	0.665	0.637
	K=200	0.717	0.681	0.679	0.708	0.663	0.664	0.646
	K=300	0.716	0.683	0.683	0.710	0.663	0.669	0.649
	K=400	0.715	0.685	0.686	0.711	0.666	0.669	0.646
	K=500	0.714	0.684	0.688	0.710	0.667	0.669	0.643
	K=600	0.714	0.686	0.691	0.711	0.668	0.675	0.643
	K=700	0.714	0.686	0.692	0.711	0.667	0.675	0.646
	K=800	0.714	0.686	0.692	0.711	0.667	0.675	0.646

6. Conclusion and Future Work

In this paper, we have proposed a new approach for text categorization based on incorporating background knowledge (WordNet) into text representation with using the χ_2 multivariate, which consists on extracting the *K* better features characterizing best the category compared to the others. The experimental results with both Reuters21578 and 20Newsgroups datasets show that incorporating background knowledge in order to capture relationships between words is especially effective in raising the macro-averaged F_1 value.

The main difficulty is that a word usually has multiple synonyms with somewhat different meanings and it is not easy to automatically find the correct synonyms to use. Our word sense disambiguation technique is not capable of determining the correct senses. Our future works include a better disambiguation strategy for a more precise identification of the proper synonym and hyponym synsets.

Some work has been done on creating WordNets for specialized domains and integrating them into MultiWordNet. We plan to make use of it to achieve further improvement.

References

[1] Bloehdorn S. and Hotho A., "Text Classification by Boosting Weak Learners Based on Terms and Concepts", *in Proceedings of the Fourth IEEE International Conference on Data Mining*, IEEE Computer Society Press, 2004.

[2] Cruse D., *Lexical Semantics*, Cambridge, London, New York, Cambridge University Press, 1986.

[3] Dash M. and Liu H., "Feature Selection for Classification", *Journal Intelligent Data Analysis*, Elsevier, vol. 1, no. 3, 1997.

[4] Green S., "Building Hypertext Links in Newspaper Articles Using Semantic Similarity", *in Proceedings of Third Workshop on Applications of Natural Language to Information Systems (NLDB'97)*, pp. 178-190, 1997.

[5] Green S., "Building Hypertext Links by Computing Semantic Similarity", *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 11, no. 5, pp. 713-730, 1999.

[6] Hofmann T., "Probmap: A Probabilistic Approach for Mapping Large Document Collections", *Journal for Intelligent Data Analysis*, vol. 4, pp. 149-164, 2000.

[7] Hotho A., Staab S., and Stumme G., "Ontologies Improve Text Document Clustering", *in Proceedings of the 2003 IEEE International Conference on Data Mining (ICDM'03)*, pp. 541-544, 2003.

[8] Ide N. and Véronis J., "Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art," *Computational Linguistics*, vol. 24, no. 1, pp. 1-40, 1998.

[9] Kehagias A., Petridis V., Kaburlasos V., and Fragkou P., "A Comparison of Word and Sense-Based Text Categorization Using Several Classification Algorithms", *Journal of Intelligent Information Systems*, vol. 21, no. 3, pp. 227-247, 2001.

[10] McCarthy D., Koeling R., Weeds J., and Carroll J., "Finding Pre-Dominant Senses in Untagged Text", *in Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pp. 280-287. Barcelona, Spain, 2004.

[11] Miller G., "Nouns in WordNet: A Lexical Inheritance System", *International Journal of Lexicography*, vol. 3, no. 4, 1990.

[12] Peng X. and Choi B., "Document Classifications Based on Word Semantic Hierarchies", *in Proceedings of the International Conference on Artificial Intelligence and Applications (IASTED)*, pp. 362-367, 2005.

[13] Pennock D., Dave K., and Lawrence S., "Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews", *in Proceedings of the Twelfth International World Wide Web Conference (WWW'2003)*, ACM, 2003.

[14] Sebastiani F., "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1-47, 2002.

[15] Voorhees E. , "Query Expansion Using Lexical-Semantic Relations", *in Proceedings of ACM-SIGIR*, Dublin, Ireland, pp. 61–69, ACM/Springer, 1994.

Zakaria Elberrichi is lecturer in computer science and a researcher at Evolutionary Engineering and Distributed Information Systems Laboratory, EEDIS at the University Djillali Liabes, Sidi-belabbes, Algeria. He holds a master degree in computer science from the California State University in addition to PGCert in higher education. He has more than 17 years of experience in teaching both BSc and MSc levels in computer science and planning and leading data mining related projects. The last one called "New Methodologies for Knowledge Acquisition". He supervises five master students in e-larning, text mining, web services, and workflow.

Abdellatif Rahmoun received his BSc degree in electrical engineering, University of Science and Engineering of Oran, Algeria, his Master degree in electrical engineering and computer science from Oregon State University, USA, and his PhD degree in computer engineering, Algeria. Currently, he is a lecturer in Computer Science Department, Faculty of Planning and Management, King Faisal University, Kingdom of Saudi Arabia. His areas of interest include fuzzy logic, genetic algorithms and genetic programming, neural networks and applications, designing ga-based neuro fuzzy systems, decision support systems, AI applications, e-learning, electronic commerce and electronic business and fractal image compression using genetic tools.