A WORDNET-BASED APPROACH TO FEATURE SELECTION IN TEXT CATEGORIZATION

Kai Zhang, Jian Sun and Bin Wang
Insititue of Computing Technology, Chinese Academy of Sciences, Beijing, China 100080

Abstract: This paper proposes a new feature selection method for text categorization. In this method, word tendency, which takes related words into consideration, is used to select best terms. Our experiments on binary classification tasks show that our method achieves better than DF and IG when the classes are semantically discriminative. Furthermore, our best performance is usually achieved in fewer features.

Key words: Feature Selection, WordNet, Text Categorization

1. INTRODUCTION

Feature selection is a key step in text classification. There are many automatic feature selection methods such as Document Frequency, Information Gain and Mutual Information. They work well in many cases. [Sebastiani 02], [Aas 99] and [Yang 97] showed their formulas and gave comparisons. Most of them use frequency of words and disregard semantic information. Their methods assume that words are equal and isolated, but actually they are different and inter-connected in semantic. WordNet ([Miller 90]), one of the best lexical databases, would be helpful to feature selection.

There are a few studies in text classification using WordNet. Rodriguez's study ([Rodríguze 97]) using WordNet synonymy, involved a manual word sense disambiguation step, and took advantage of Reuters topical headings. Finally they got good results. Sam Scott's work ([Scott 98]) used hypernyms, and their representation works well for extended or

unusual vocabulary like songs, but it is not likely to work well for texts that are concisely and efficiently written such as the texts in Reuters-21578. Word sense disambiguation is the bottleneck to his work, because it is difficult to select the correct meaning of a word in context. For feature selection, word sense disambiguation is not a sticking point. That's why we use WordNet to improve feature selection.

In this paper, we will study how to use WordNet to extract good features. The paper proceeds as follows. Section 2 discusses the usage of WordNet in feature selection. In section 3, we present the method of feature selection. Section 4 shows the experimental results and comparison. In section 5, we give comparisons with related work such as LSI and clustering of words. Finally section 6 presents the conclusions and future work.

2. HOW TO USE WORDNET

In WordNet, English words have been organized into synonym sets (synsets), and each synset represents one underlying lexical concept. A word may have several meanings, corresponding to several synsets. There are different types of relations between synsets such as antonymy, meronymy and hyponymy. Nouns in WordNet are organized as a lexical inheritance system, hyponymy generates a hierarchical semantic organization, and meronymy is also implemented in the noun files. These relations are fundamental organizing principles for nouns, and useful for text mining.

Suppose we are trying to classify documents of two classes, for example, gold and livestock, given some samples for training. We do not know the topics of the classes. Without computer, which features can we choose from the training documents? It will be commonly observed that some similar words mostly occur in one set. These words are commonly good features. For figure 1, "gold", "silver", "mine", "beef", "hog" are usually good features. "Company" is not likely to be a good semantic feature, because his semantic neighbor "market" occurs frequently in both sets. Thus we can discover:

Good features tend to be gathered in semantics, especially the words that relate to the class topics. So similar words that mostly distribute in one training set would be good features.

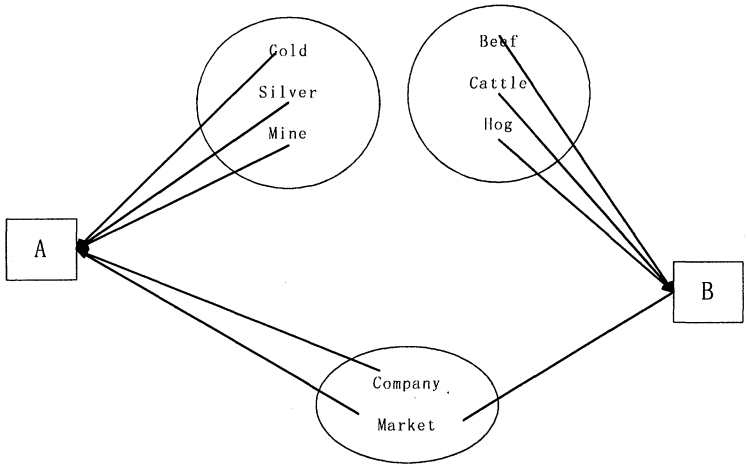

Figure 1: Two classes and the semantically clustered words.

Our method is to simulate this way. There are two semantically different classes: A and B. It is hard to find semantically related words directly, for there is no direct relation between words in WordNet. So we convert word frequencies to synset densities, which are the assumed frequencies of the synsets in training set.

Now we can interpret our method in another way. For each synset, it has two densities: density_A and density_B. If density_A is much greater than density_B, we can say the synset is tendentious to A. The formula *density_A/(density_A+density_B)* can be used to evaluate the tendency. However, as described above, a discriminative synset in semantics often has neighbors of the similar tendency. It would be better to take into account the neighbors of the synset. We cluster the synsets, and compute the total densities of the synsets in each cluster as the cluster densities of synsets in the cluster.

The formula *cluster_density_A/(cluster_density_A+cluster_density_B)* is more suitable to evaluate the tendency of synsets. Then the tendencies of synsets are converted to word tendencies. Finally DF is considered in the weight formula to remove infrequent words. All detailed formulas will be given in Section 3.

Figure 2 gives an example. The four words and their occurrences in both classes are shown at bottom. Synset and their relations are shown in the graph, types and directions of the relations are ignored. Densities are computed according to word occurrence. Black stands for synset density of A, while white stands for that of B. We notice that the only synset of Word3 has many black neighbors. Word3 would be a good feature. Word2 has 2 meanings: one is in a cluster of almost black, while the other is in that of half

black. We can conclude Word3 has more semantic tendency than Word2, though Word2 does not occur in the training set of B and has more occurrences than Word3 in the training set of A. We can also conclude Word1 and Word4 are usually improper features.

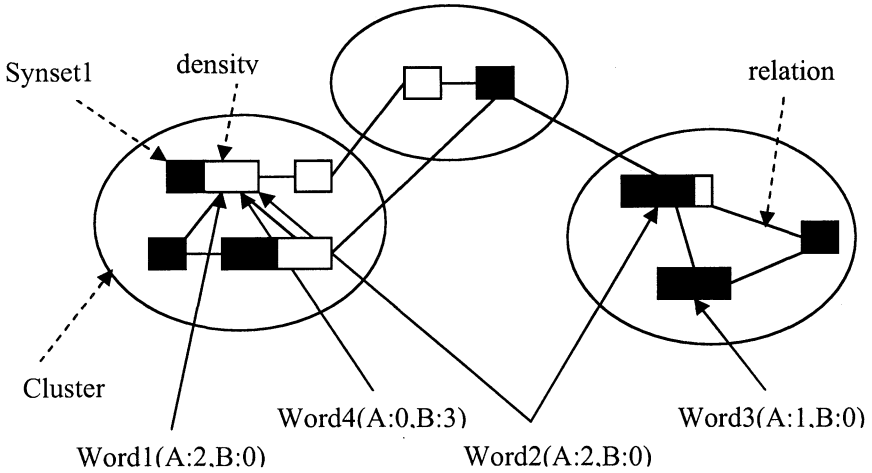

Figure 2:Clustering of synsets

3. FEATURE SELECTION USING WORDNET

This section gives detailed algorithm of the feature selection method. First, we process the documents to get noun occurrences, which are the most useful information. Then as described in the previous section, we compute the density, cluster the synsets and give the weighting formula.

3.1 Preprocessing

In preprocessing, we use Eric Brill's part of speech tagger ([Brill 92]) to tag the training and test documents. It is a simple rule-based part of speech tagger. We only use nouns to select features. Verbs and other words are discarded, because they render less semantic information. The nouns are stemmed to their origin form. Finally, we remove the stop words and words that have more than 5 meanings. These words are commonly bad features.

3.2 Computing density

In WordNet, a word may have several meanings. Without word sense disambiguation, its frequency should be shared by its synsets equally. So the density of synset s should be summary of shares of several word frequencies. The "density" of a synset s corresponding to class c is given by:

$$density(s,c) = \sum_{w \in s} \frac{word\_frequency(c,w)}{Synset\_count(w)}$$

$$= \sum_{w \in s} \frac{times(c,w)}{synset\_count(w) \bullet word\_count(c)}$$

Times(c, w) denotes the occurrence of word w in the training documents of class c. Synset_count (w) is the count of synsets containing w in WordNet. Word_count(c) is the count of nouns in the training documents of class c. Synset density defined in this paper is similar to that of [Scott 98].

3.3 Clustering

Here we use one of the simplest clustering methods described as below. In the algorithm, the distance of two synsets is defined as the length of the shortest path between them in WordNet.

Input: The threshold T and N synsets
Output: Clusters of synsets

1. Arbitrarily choose a synset as a seed
2. Repeat
   Arbitrarily choose a synset (s) that is not clustered.
   Find the nearest seed (s') of s
   If the distance >T, let s be a seed
   Else s is put into the cluster of seed s'.
   Until all synsets are clustered.

We choose this method because it is simple. We don't think more complex methods will certainly get better result. It is difficult to achieve the most suitable method. This will be our future work to achieve more suitable methods.

3.4 Computing weights

After clustering, the density of a cluster r to class c is defined as:

$$cluster\_density\ (r,c) = \sum_{j \in r} density\ (j,c)$$

The cluster density can be regarded as the occurrence of some similar meaning in the cluster r in the training set of class c. Furthermore, we will have to compute word tendency to evaluate in which test set the word is likely to occur .The tendency should be decided by all the word meanings (synsets), but we use cluster density because of the hypothesis in Section 2.

The tendency of a word w to class A is defined as:

$$t(w) = \frac{\sum_{i \in Synset(w)} cluster\_density(cluster(i),A)}{\sum_{i \in Synset(w)} cluster\_density(clsuter(i),A)+cluster\_density(cluster(i),B)}$$

$$= \frac{\sum_{i \in Synset(w)} \sum_{j \in Cluster(i)} density(j,A)}{\sum_{i \in Synset(w)} \sum_{j \in Cluster(i)} density(j,A)+density(j,B)}$$

Where synset(w) denotes the synsets containing word w. Cluster(i) is the cluster of synset i. When t(w)=1, w is likely to semantically related to A. When t(w)=0, w is likely to semantically related to B. Words that have tendency of approximate 0 or 1 will be good choices for features.

For some reasons, some words have only one synset, and the synset is in the cluster of its own. They often gain good tendency, but they are not suitable for features.. To be a good feature, word frequency must be high .The final weight formula is given by:

$$WN(w) = |\, t(w) - 0.5\,| * \log(\min(DF(w), N) + 1)$$

Where DF (w) is the document frequency of the word w. N is the maximum document count of two classes. WN (w) is the final weight formula. Document Frequency would be a good parameter, and DF above N will be truncated to N. We use *log(DF+1)* here, which is like the formula of TFIDF.

WN(w) can be the final formula. But to balance the features between A and B, interleaving can be done in feature sequence. Here is a sample of interleaving:

Original: A1,A2,A3,B1,B2,A4,B3,B4

Final: A1,B1,A2,B2,A3,B3,A4,B4

4. EXPERIMENTS AND RESULTS

The classification tasks used in this study are drawn from three different pairs of classes in Reuters-21578. We do experiments using Rainbow, a good classifier with many options. We compared our method with IG and DF, which are regarded as two of the best feature selection methods. We use three pairs of classes ([gold/livestock], [crude/trade], [coffee/ship]), like that of Scott's experiments. For each pair of classes, we divide the documents for 20 times. For each division, we randomly choose 60 documents for training (30 for each class), and the others are documents for testing. We also have the option of feature count from 5 to 600 and DF threshold from 0 to 60. We use rainbow to get accuracy on different tasks and options.

First, table 1 gives the top 10 selected features in one classification in task of [gold/livestock], order by IG or WN (interleaved). We notice that features selected by WordNet are more semantically related to the class topics. "Company" perhaps is a proper feature for term frequency, but it is not directly related to the topics. From the relative small training sets, we can hardly draw the conclusion that "company" seldom occurs in the test set of livestock.

Table 1: Features selected by IG and WordNet

	IG	**WN**
1	Gold	Gold
2	Ounces	Beef
3	Ounce	Mine
4	Mine	Pork
5	Ton	Ounce
6	Cattle	Cattle
7	Lt	Ore
8	Company	Farm
9	Mining	Tons
10	Reserves	Hog

Second, Figure 3 gives the comparison between IG and WN in the task of [gold/ livestock]. The Y-coordinate is the average accuracy from 20 tests of different division, while the X-coordinate is the feature count. From the figure, we can see that WN get higher accuracy at fewer features. On the curve of WordNet, the vertex (99.46) is at the feature count of 10. The leftmost point shows the accuracy (99.04) at the feature count of 5. We can naturally draw the conclusion that our top 10 features are more discriminative than those of IG in this task. With the increment of feature count, the two curves almost converge like the curves in Yang's experiments ([Yang 97]).

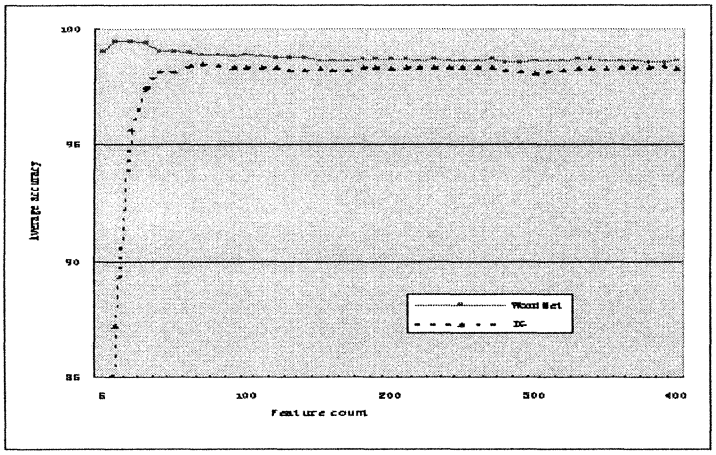

Figure 3: Comparison between IG and WordNet in task of gold/livestock

Table 2: The best error rates in different tasks

Tasks	Balance	IG	DF	WN
Gold/Livestock	134/113	1.52	0.69	0.53
Crude/Trade	626/543	2.66	3.07	4.33
Coffee/Ship	158/133	2.66	2.19	1.04

From Table 2,"Balance" refers number of examples in each class. "IG", "DF", "WN" show to the best error rates in different feature numbers. We can see that WordNet works well in task of [gold/livestock] and [coffee/ship]. While in Scott's experiments ([Scott 98]) WordNet does not improve the accuracy in task of [gold/livestock]. These classes are more specific, but for more general classes like crude and trade, it does not work

well. The reason is that they have more semantic intersections, for example, the cargo may be oil and ore, which would be a little difficult for WordNet. It will be our future work to improve the accuracy of this kind of task.

5. DISCUSSIONS

We achieve better results when using fewer features. Compared with other dimensionality-reduction methods such as LSI and cluster of words, it is worth noting that our feature space is still on English words. It is difficult to get higher accuracy in fewer word-features, because a feature in LSI or Cluster of words can be affected by a lot of words, a word-feature can be affected by only one word. Ten word-features perhaps do not occur in some test documents, but we get high accuracy in such risk.

Furthermore synset-feature is discussed in Sam Scott's papers ([Scott 98]). It is difficult to choose from synset-feature and word-feature. For synset-feature, it is important to get the right synset of words. However, the accuracy is not very high. For example, in Li's paper [Li.95], the accuracy is only about 72%. Without word sense disambiguation, experiments show that synset-feature is not found to produce significant performance. So word sense disambiguation will play an important role in text classification.

6. CONCLUSIONS AND FUTURE WORK

Given the results from section 4, we can conclude that WordNet is a good resource for text classification. In this paper, we take the advantage of the related words using WordNet. Terms that have same tendencies as its neighbors will be good features. Based on this hypothesis, we use WordNet to extract more semantically related features. And this method can improve the accuracy in tasks that have semantically distinct classes.

However, as many people believed, significant advances must be made before NLP techniques can be used to improve text classification. Experiments shows that this method does not tend to perform well on classification tasks involving broadly defined or semantically related classes. This is a tentative and ongoing work. Clustering method and functions in this approach can be improved by using the WordNet hierarchy. And it would be helpful to incorporate other feature selection methods such as IG. For instance, formula based on WordNet can be given to tell which one is more suitable to certain classification task. And we will evaluate this method on

more tasks to get more informative results. Multi-class tasks are also interesting.

Furthermore, we can also find latent features by using WordNet clusters. This perhaps works well for tasks that have smaller training sets.

REFERENCES

1. [Aas 99] Kjersti Aas, Line Eikvil.Text Categorisation: A Survey. Technical report,Norwegian Computing Center, June.
2. [Baker 98] L. Douglas Baker, Andrew Kachites McCallum. Distributional clustering of words for text classification. In Proc. SIGIR-98, Melbourne, Australia 1998.
3. [Blum 98] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training.In Proceedings of COLT'98,1998
4. [Brill 92] Eric Brill. A simple rule-based part of speech tagger. In proceedings of Third Conference on Applied Natural Language Processing, ACL, 1992
5. [Li.95] Xiaobin Li, Stan Szpakowicz and Stan Matwin. A WordNet-based Algorithm for Word Sense Disambiguation. In Proc. IJCAI_95, Montréal, Canada, 1995
6. [Miller 90] George A. Miller. WordNet: an On-line Lexical Database. International Journal of Lexicography 3(4), 1990
7. [Rodríguze et al.97] Manuel de Buenaga Rodríguze, Jos María Gómez-Hidalgo and Belén Díaz-Agudo. Using WordNet to Complement Training Information in Text Categorization. In Proc. RANLP-97, Stanford March 25-27,1997
8. [Scott 98] Sam Scott, Stan Matwin (1998). Text Classification Using WordNet Hypernyms. In S. Harabagiu & J. Yue Chai (Eds.) *Usage of WordNet in Natural Language Processing Systems: Proceedings of the Workshop*(pp. 45-52).
9. [Sebastiani 02] Fabrizio Sebastiani. Machine Learning in Automated Text Categorization. ACM Computing Surveys Vol. 34 ,No.1 March 2002 .
10. [Yang 95] Yiming Yang, Noise reduction in a statistical approach to text categorization. In Proc. SIGIR-95 (Seattle,WA,1995)
11. [Yang 97] Yiming Yang , Jan O.Pedersen. A comparative study on feature selection in text categorization. In Proc. ICML-97, Nashville, TN, 1997
12. [Yang 99] Yiming Yang , Xin Liu .A re-examination of text categorization methods. In Proc. SIGIR-99, Berkley, CA, 1999