

Combining Statistics and Semantics via Ensemble Model for Document Clustering

Samah Jamal Fodeh
Michigan State University
3115 Engineering Bldg
East Lansing, MI, 48824

fodehsam@msu.edu

William F Punch
Michigan State University
3115 Engineering Bldg
East Lansing, MI, 48824
517- 3533541

punch@msu.edu

Pang-Ning Tan
Michigan State University
3115 Engineering Bldg
East Lansing, MI, 48824
517- 4329240

ptan@msu.edu

ABSTRACT

Incorporating background knowledge into data mining algorithms is an important but challenging problem. Current approaches in semi-supervised learning require explicit knowledge provided by domain experts, knowledge specific to the particular data set. In this study, we propose an ensemble model that couples two sources of information: statistics information that is derived from the data set, and sense information retrieved from WordNet that is used to build a semantic binary model. We evaluated the efficacy of using our combined ensemble model on the Reuters-21578 and 20newsgroups data sets.

Keywords

WordNet, ensemble learning, text clustering, disambiguation.

1. INTRODUCTION

The rapidly growing availability of large tracts of textual data such as online news feeds, blog postings, emails, and discussion board messages, has made the need for improved text clustering an important current research area. However, despite the extensive research, clustering unstructured, textual information remains a challenging problem. For example, the nature of the unstructured textual information makes it hard for current clustering algorithms to capture the intrinsic structure that we desire [3]. Individual data sets also have unique characteristics which add more complexity to mapping or deciding upon the clustering methodology that works best for a particular data set. Moreover, the lack of labeled examples in unsupervised clustering make the partitioning task an ill-posed problem since there is no adopted methodology well-known to produce the ideal clustering [3]. To overcome these challenges, researchers have begun to investigate alternative clustering approaches that incorporate **background knowledge** to guide each partitioning task and thus alleviate the difficulty of finding a single, best approach [2][6].

One way to add background knowledge is through semi-supervised clustering [1], where the domain information is

provided in the form of labeled examples or must-link (ML) and cannot-link (CL) constraints. This explicit information suggests the availability of an expert in the domain who would annotate the labels of the documents or summarize the important associations between documents in the data set. In practice, this human intervention can be expensive and could produce inaccurate results depending on the reliability of the information provided. To address these problems, there has been some recent work that attempts to incorporate background knowledge into the partitioning task without any user or expert dynamic interaction [8]. Unlike semi-supervised clustering, this kind of background knowledge provides general information about the relationship between the features and is applicable to any data set with similar types of features. For document clustering, recent interest has focused on incorporating contextual knowledge in the form of linguistic ontologies into clustering algorithms such as WordNet [2][6][8]. WordNet is an ontology that includes not only the senses of the words, but also their relationships with each other. WordNet addresses the synonymy and polysemy problem of text documents by replacing the words by their most appropriate sense as used **in the context of the document**.

In this paper, we investigate the effectiveness of combining term statistics with semantic knowledge acquired from WordNet to improve document clustering. Our analysis shows that a straightforward replacement of the words by their corresponding senses from WordNet may not always improve the clustering results. This is because the clustering algorithm must deal with issues such as the increasing dimensionality of the data (when the nouns are replaced by their corresponding senses) and noise (when incorrect senses are selected as features). We propose to address these issues using a compound ensemble clustering algorithm that combines the statistics information from the data with the sense information from WordNet. Our approach for combining the models takes into account the consistency of each clustering solution and is applicable to any clustering algorithms (including k-means). We evaluated the effectiveness of our method using two benchmark datasets: Reuters-21578 and 20newsgroups. Our experimental results suggest that the proposed method helps to improve the clustering results significantly when applied to data sets where sense information is valuable to disambiguate words that are used in multiple contexts.

2. Semantic Similarity using WordNet

WordNet is a hierarchical linguistic ontology that groups words into synsets. Each synset defines a certain concept. Synsets are

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SAC '09, March 8-12, 2009, Honolulu, Hawaii, U.S.A.

Copyright 2009 ACM 978-1-60558-166-8/09/03...\$5.00.

linked by semantic relations such as hypernym and hyponym, indicating class-subclass relationships between the synsets. Several semantic distance measures can be used to compute the semantic similarity between two synsets in the Wordnet hierarchy. In particular, we used the Wu-Palmer measure, which utilizes information such as sense depth. Wu-Palmer computes the similarity between two senses by finding the least common subsumer (LCS) node that connects their senses. The LCS of two senses, s_p and s_q , is the lowest intersecting node between the paths of s_p and s_q from the root of the WordNet class-subclass hierarchy. Once the LCS has been identified, the Wu-Palmer distance is given by the following equation:

$$\sigma_{\text{Wu-Palmer}}(s_p, s_q) = \frac{2d}{L_p + L_q + 2d} \quad (1)$$

where d is the depth of the LCS from the root, L_p is the path length between s_p and the LCS, and L_q is the path length between s_q and the LCS.

A key step in our proposed approach for incorporating semantic knowledge from WordNet is to identify the most appropriate sense associated with each noun in a given document. Our approach is based on the assumption that the sense of a noun is determined by the context in which it is being used in the document. For example, consider the word ‘‘cat’’, which has eight meanings as a noun in WordNet. If it is used in a document that contains other words such as ‘‘kitten’’, ‘‘Persian’’, and ‘‘pet’’, we expect its sense refers to the ‘‘feline mammal’’ sense of cat, and not one of the other seven (such as a farm machine or a particular type of X-ray). However, if the word ‘‘cat’’ appears in a document that contains other nouns such as ‘‘construction’’ and ‘‘builder’’, its sense most likely refers to a ‘‘Caterpillar’’, which is a large tracked vehicle used for moving earth in construction.

Let $S_i = \{s_{i1}, s_{i2}, \dots, s_{ik}\}$ denote the set of all senses associated with the noun t_i according to the WordNet ontology. Given a document d , we determine the most appropriate sense \hat{s}_i of a noun t_i by computing the sum of its similarity to other noun senses in d , i.e.:

$$\hat{s}_i = \arg \max_{s_{ii} \in S_i} \sum_{t_j \in d} \left[\max_{s_{jm} \in S_j} \sigma(s_{ii}, s_{jm}) \right] \quad (2)$$

where $\sigma(s_p, s_q)$ is the WordNet similarity between two senses, s_p and s_q . Furthermore, since the senses of a given noun in the WordNet hierarchy are arranged in descending order according to their popularity, we restrict our consideration to the top 3 senses of each given noun.

Once the appropriate senses have been chosen for the nouns, we transform each document vector of terms into a binary vector of their corresponding senses.

3. ENSEMBLE CLUSTERING

The proposed ensemble clustering framework combines the clustering solutions obtained from the semantic similarity of the documents (**semantic binary model**) with those obtained based on frequency similarities (**nouns frequency model**). Our rationale for using ensemble clustering is that, although individual clustering solutions (using either frequency or semantic similarity) may make poor decisions regarding the cluster assignments for some documents, one may be able to improve clustering results by

considering their collective decisions¹. The proposed framework is also highly flexible because it may accommodate any baseline clustering algorithms as well as methodology for creating different instances of the ensemble.

The ensemble clustering framework uses two types of data inputs: (1) a document-noun frequency matrix \mathbf{A}_f , and (2) a document-sense binary matrix \mathbf{A}_s . A set of frequency-based clustering solutions, $M_{\text{Fensemble}}$, is generated from \mathbf{A}_f using the methodology presented in Section 3.1. Analogously, a sense of sense-based clustering solutions, $M_{\text{Senseable}}$, is obtained by applying the methodology given in Section 3.2. The clustering solutions from $M_{\text{Fensemble}}$ and $M_{\text{Senseable}}$ are then aggregated to obtain the final clustering using the approach described in Section 3.3.

3.1 Noun Frequency Model

The noun frequency model is generated from applying ensemble clustering to the document-noun frequency matrix \mathbf{A}_f . First, we randomly sample a subset of the nouns, $V_s \in V$. To alleviate bias in the sample size $|V_s|$, the number of nouns to be sampled is an integer chosen randomly between $m/2$ and $m-1$, where m is the total number of nouns in the original dataset. A truncated document-noun frequency matrix, \mathbf{A}_f' is then created by choosing only the nouns in each document that belongs to the subset V_s . Once the truncated matrix is obtained, the weights for each document vector are further normalized using the TFIDF method. Finally we apply the standard k-means algorithm to obtain an $N \times k$ frequency-based cluster membership matrix, C_f , whose (i,j) th element is equal to 1 if the document d_i belongs to cluster j and 0 otherwise.

3.2 Semantic Binary Model

Our approach for creating the semantic binary model is quite similar to the approach described in the previous section. However, instead of using the document-noun frequency matrix, we use the document-sense matrix as input to the ensemble clustering algorithm. The sample size is a random integer chosen between $m/2$ and $m-1$. A truncated document-sense binary matrix \mathbf{A}_s' is then created by removing all the senses not selected by the sample. After normalization using the TFIDF method, we apply the k-means clustering algorithm to obtain the sense-based cluster membership matrix, C_s .

3.3 Combined Ensemble Clustering

This section describes our proposed approach for combining the noun frequency model with the semantic binary model. First, an $N \times N$ weighted co-association matrix S_f is computed from the set of frequency-based cluster membership matrices in $M_{\text{Fensemble}}$. The co-association matrix represents the number of times a pair of documents is assigned to the same cluster in the ensemble, weighted by the ‘‘quality’’ of the individual clustering solution. Formally it is iteratively computed from the frequency-based cluster membership matrices C_f as follows:

$$S_f^{(t+1)} = S_f^{(t)} + w_t C_f^{(t)} C_f^{(t)T} \quad (3)$$

where the matrix product $C_f^{(t)} C_f^{(t)T}$ is a binary 0/1 matrix that indicates whether a pair of documents belongs to the same cluster

¹ Assuming each clustering solution is independent and is doing better than random cluster assignments.

during the t^{th} iteration of the ensemble and the weighting factor w_t measures the quality of the clustering. The matrix product $C_f^{(t)}C_f^{(t)T}$ is also known as an incidence matrix in clustering literature. The weighting factor w_t is then computed by correlating the cosine similarity matrix between each pair of document with the incidence matrix $C_f^{(t)}C_f^{(t)T}$. The higher the correlation is, the greater the level of agreement between the clustering results and the document similarity matrix. Equation (3) is iteratively updated using all the clustering solutions from the noun frequency model. The weighted co-association matrix S_f effectively encodes the likelihood that a pair of documents is in the same cluster based on its term frequency information. It is certainly possible to apply a clustering algorithm such as k-means on S_f to produce a final clustering for the noun frequency model. Similarly, we may repeat this procedure to obtain a sense-based weighted co-association matrix S_s :

$$S_s^{(t+1)} = S_s^{(t)} + w_t C_s^{(t)} C_s^{(t)T} \quad (4)$$

where the incidence matrix $C_s^{(t)}C_s^{(t)T}$ depends on the clustering solutions of the semantic binary model and the weighting factor w_t is the correlation coefficient between the cosine similarity of documents (computed from the document-sense matrix A_s) and the incidence matrix. The overall semantic binary model can be obtained by applying the kmeans algorithm to the weighted co-association matrix. However, since our intention is to combine the noun frequency model with semantic binary model, we may aggregate their weighted co-association matrices as follows:

$$S_{\text{Combined}} = \alpha S_f + (1 - \alpha) S_s \quad (5)$$

where α is a parameter that governs the tradeoff between using both models. We will apply a clustering algorithm such as kmeans on the combined weighted co-association matrix to obtain the final clustering results.

4. EXPERIMENTS

Selecting a data set to test our approach upon has to be done carefully. We wanted to work with a data set that reveals the power of substituting the noun by its sense. So testing with a dataset that is rich with vocabulary used to express the same meaning, or vocabulary that has multiple meanings would give our algorithm a better chance to demonstrate the effect of incorporating the semantic knowledge from WordNet. Two benchmark data sets were selected for our experiments—the Reuters-21578 and 20newsgroups data sets.

For *Reuters-21578 dataset* we selected only documents that belong to the top 20 largest categories for our experiment. We then sampled at most 200 documents from each category. The final size of our text corpus was 2655 documents.

For *20newsgroups dataset*: we aggregated the original training set into 6 distinct categories. For example, we aggregate categories such as rec.autos, rec.motorcycles, rec.sport.baseball, and rec.sport.hockey into one class. The original data was divided into 60% training set and 40% test set. For our experiments, we sampled 6000 articles from the training set and applied our algorithm to detect the 6 distinct categories.

All the experiments conducted in this study employ kmeans as the baseline algorithm. The number of clusters is equal to the number

of categories in the original data. Since kmeans is sensitive to the choice of initial centroids, we repeat each experiment 50 times and report their average entropy or purity. We compared our method against Latent Semantic Indexing (LSI) which is a statistical method for identifying the latent structure in a set of documents by analyzing the relationships between the documents and their corresponding terms.

Table 1: Intra-Document Clustering for Reuters Data Set.

Document	Clusters
1	call
	rates, loan
	screen
	degree, monetary_value
	department , militia
	corn, grain, barley, wheat, oat
	liberation ,agribusiness
	average, iodine
	national, substitute ,sorghum
	hundredweight,two, three, four,six
2	tallow
	stallion, Canadian
	authority, agreement
	purchase, bargain
	department, point, Honduras
	transshipment, agribusiness
	corn, wheat
	port, screen, ship
	measure , metric_ton
3	board
	estimate, intervention, beginning
	metric_ton , season
	left , French
	barley, cereal, wheat, corn
	prognosis , manner_of_speaking

4.1 Intra Document Sense Disambiguation

This experiment aims to demonstrate the effectiveness of our method in selecting the most appropriate sense to substitute for a noun in the context of a particular document. One way to do this is to cluster the senses and examine the resulting clusters. If we observe distinctive topics in the clusters then this indicates that the selected senses were able to reveal some of the semantic content of the document. In order to apply clustering within the documents, we built a similarity matrix for the senses using Wu-Palmer similarity measure [10]. Intra-document clustering is then performed using the complete-link agglomerative hierarchical clustering algorithm.

Table 1 shows the results of clustering three documents that belong to the grain category in the Reuters dataset. Document 2 shows well-related semantic clusters such as (purchase, bargain), (corn, wheat), and (measure, metric_ton). These clusters represent the different semantic concepts that are important in the document. Nevertheless, we do observe some impure clusters using the senses that were chosen. For example, in document 1, although we found a closely-related semantic cluster (corn, grain, barley, wheat, oat), we also missed a sense (sorghum) that should be added to this cluster. Instead, sorghum was added to a cluster that does not have a high semantic similarity with gain.

These results suggest that our technique for fixing the sense of a word using WordNet was quite successful to form a group of semantically related senses. We therefore argue that this method was able -to a good extent- to separate the different topics in the document which should improve document clustering results.

4.2 Document Clustering using Senses

After fixing the sense of each word using the approach described in Section 2, we transform each document vector into a binary vector of their senses. For the Reuters dataset, we started with 5922 nouns as features. After applying WordNet, the data is transformed into a document-sense matrix with 6559 senses. Likewise, in the 20newsgroups dataset, our method converts the 13801 nouns into 14589 senses. Table 2 shows the number of features used for the different approaches (including LSI).

Table 2. Number of features used

Dataset	Noun-frequency model	Semantic binary model	LSI
20newsgroups	13801	14589	34757
Reuters	5922	6559	18238

The number of features has increased after word sense disambiguation since the algorithm may resolve the same noun into multiple senses depending on the context of the documents. Next, we apply k-means clustering on both the document-noun frequency matrix A_f and document-sense binary matrix A_s . Table 3 shows the results of the clustering, which are based on the average entropy for 50 iterations of applying kmeans with different initial centroids. These results however do not seem to justify the need to replace the nouns with their corresponding senses. In fact, the use of sense information seems to degrade the entropy significantly for the Reuters data set.

Table 3. Comparison of average entropy for K-means

Dataset	Noun-frequency model	Semantic binary model
20newsgroups	0.86	0.88
Reuters	0.97	1.19

One possible explanation for the poor results is due to increasing number of features when we replace the nouns by their senses. To investigate the effect of the number of dimensions, we have performed Singular Value Decomposition (SVD) on both the document-noun frequency matrix and document-sense binary matrix prior to applying the kmeans algorithm. The former is almost equivalent to a Latent Semantic Indexing (LSI) except we had removed the adjectives, verbs, and nouns that are not registered in WordNet. Figure 1 shows the results of applying k-means clustering on the document-noun frequency matrix, document-sense binary matrix, and LSI.

For the 20newsgroups data, the entropy values improve significantly after incorporating sense information. Nevertheless, these values are still worse than the results without applying SVD (see Table 3). For the Reuters data, the entropy values for clustering using sense information are generally worse than those obtained by applying SVD on the nouns and the LSI method. The results provided in this section suggest that transforming the nouns into senses alone is insufficient to improve the clustering results. In fact, the results may be worse due to the increased dimensionality of the data or when the wrong sense is chosen to replace some of the nouns.

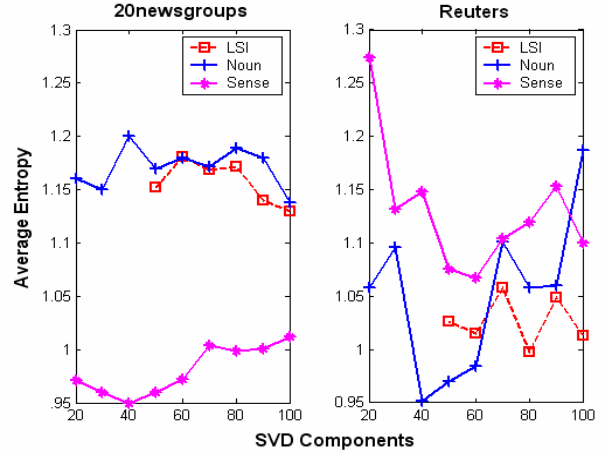


Figure 1. Comparison of Avg entropy for K-means with SVD

Nevertheless, we still expect some documents to be correctly placed in the right cluster because of the word sense disambiguation. What is needed is a clustering algorithm that: (1) better utilizes the sense information in combination with the term statistics information. (2) deals with the increasing number of dimension when replacing the nouns by their senses. (3) tolerant to noise when incorrect senses are used. Because of its flexibility and robustness, we conjecture that ensemble clustering is an appropriate approach for combining term statistics with semantic knowledge for document clustering.

4.3 Ensemble Clustering

In this work, we combine the clustering results from both the semantic binary model (denoted as M_{Sense}) and the noun frequency model (denoted as M_{Noun}) into one final clustering (denoted as $M_{ensemble}$). Our motivation for using ensemble clustering is because of its flexibility to accommodate any input data matrix (either the document-noun frequency matrix, the document-sense binary matrix, or both), its ability to deal with high dimensionality via feature subsampling, and its resilience to noise and other variability in the data.

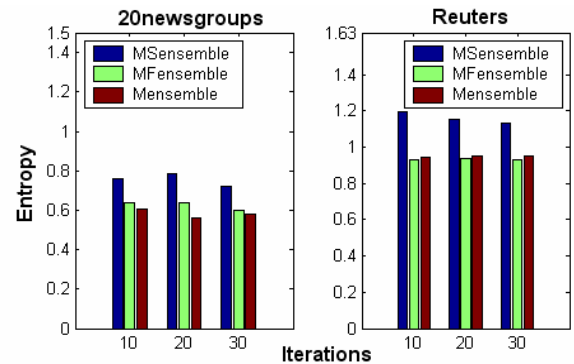


Figure 2. Comparison of entropy for ensemble clustering

Figure 2 shows the results of applying the three ensemble clustering methods to the Reuters and 20newsgroups data. The number of iterations for the ensemble in this experiment is varied from 10 to 30 runs. For each number of iterations, we combine half of the clustering results from M_{Sense} with another half from M_{Noun} to generate the final clustering $M_{ensemble}$. As mentioned in Section 3.3, each clustering solution in the ensemble will be

weighted according to the quality of their clusters (which is measured in terms of the correlation between the cosine similarity of the documents and the resulting incidence matrix of the clusters). We observed that the weighting factors w_i associated with the solutions in $M_{Fensemble}$ were generally greater than the weighting factors for $M_{Sensemble}$. For example, for the Reuters dataset the average of the weights of the $M_{Fensemble}$ was approximately .5 compared to .3 in the $M_{Sensemble}$.

This observation is consistent with our results in Section 4.2 where the noun frequency model appears to produce better clusters than using the semantic binary model. The weighted clusters in each ensemble were aggregated in a co-association matrix (with $\alpha = 0.8$) that reflects the consensus of the individual runs on allocating the documents across the clusters. Figure 2 shows a comparison of the entropy values between the three ensembles for both data sets. For the 20newsgroups dataset, the compound ensemble $M_{ensemble}$ achieved the lowest entropy score. For example, after 20 iterations, the entropy score for the compound ensemble is considerably lower (.559) than the scores for the noun frequency model (.636) and semantic binary model (.787). This result suggests that our compound ensemble method is capable of enhancing the clustering results by taking advantage of the variability in the clustering solutions obtained from the term statistics and semantic knowledge. Even though the semantic binary model has higher entropy than noun frequency model, it still provides useful solutions that can be exploited by our compound ensemble. Furthermore, all the ensemble clustering results are significantly better than the results for individual runs (Section 4.2) even though it uses only a sample of the original features. In the Reuters data set, no significant improvement was observed for the compound ensemble $M_{ensemble}$ over the frequency ensemble $M_{Fensemble}$. This result suggests that the nature of the data set also plays a significant role in determining the effectiveness of incorporating semantic knowledge from WordNet.

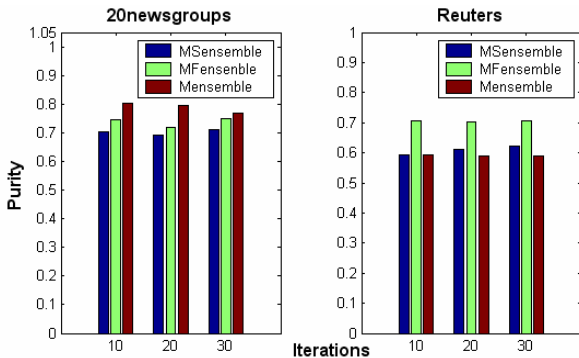


Figure 3. Comparison of purity for ensemble clustering

Figure 3 shows the purity values for the different ensemble clustering methods. Once again, the compound ensemble achieved the highest purity score (.802) for the 20newsgroups data set whereas the noun frequency ensemble model has the highest purity for the Reuters data set. Finally, it is worth noting that our proposed method using the compound ensemble clustering significantly improved the clusters quality compared to LSI for both datasets. The lowest entropy obtained for the Reuters in LSI was 1.01 using 100 components compared to .947 after 10 iterations using our compound ensemble method. For the 20-newsgroups dataset, LSI achieved an entropy of 1.13 with 100

components compared to .559 when applying the compound ensemble at 20 iterations.

5. CONCLUSION

This paper presents a methodology for combining term statistics with semantic knowledge from WordNet for document clustering. Our analysis shows that a straightforward replacement of the words with their senses may not necessarily improve the clustering results, which is consistent with some of the previous results reported in [6] and [8]. The clustering algorithm needs to be flexible and robust enough to deal with the higher dimensionality and noise due to improper selection of senses. To overcome these challenges, we propose an ensemble clustering method that systematically combines clustering solutions from a noun frequency model with those from a semantic binary model based on the consistency of their clustering solutions. Our experimental results show that the ensemble method is effective on some but not all data. We are presently doing further research to better determine what characteristics of the data are most suitable for the approach.

6. ACKNOWLEDGMENT

We acknowledge the support of the National Archives and Records Administration and the MATRIX group at Michigan State University.

7. REFERENCES

- [1] Bradley P., Bennett K., and Demiriz A., Constrained k-means clustering. *Microsoft Research Technical Report, MSR-TR-2000-65*, 2000.
- [2] Hotho A., Staab S., Stumme G, WordNet improves text document clustering. In *Proc. of the SIGIR 2003 Semantic Web Workshop*, 2003, 541-544.
- [3] Goe J. , Tan P. N., and Cheng H., Semi-supervised Clustering with Partial Background Information. In *Proc. of SIAM Int'l Conf on Data Mining*, Bethesda, MD 2006.
- [4] Mann H. B., Whitney D. R. On a test whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18, 1947, 50-60.
- [5] Miller J., WordNet: a lexical database for English, Communications of the ACM. 1995.39-41
- [6] Sedding J., Kazakov D., WordNet-based text document clustering. In *Proc. of the 3rd Workshop on Robust Methods in Analysis of Natural Language Processing Data*. 2004 , 104-113
- [7] Steinbach M. and Karypis G. and Kumar V., A comparison of document clustering techniques. In *proc. of KDD Workshop on Text Mining*, 2000.
- [8] Termier A., Rousset MC, Sebag M, Combining statistics and semantics for word and document clustering, In *Proc. of IJCAI*, 2001, 49-54.
- [9] Topchy A., Jain A.K., Punch W., A mixture model for clustering ensembles, In *Proc. of SIAM Conference on Data Mining*, 2004, 379-390.
- [10] Wu Z. and Palmer M. Verb Semantics and Lexical Selection. In *Proc. of the 32nd Annual Meeting of the Assoc. for Computational Linguistics*, 1994, 133-138.