Dimensionality Reduction of very large document collections by Semantic Mapping

Renato Fernandes Corrêa, Teresa Bernarda Ludermir
Center of Informatics, Federal University of Pernambuco
P.O. Box 7851, Cidade Universitária, Recife - PE, Brazil, 50.732-970
email: {rfc, tbl}@cin.ufpe.br

Keywords: Document Clustering, Dimensionality Reduction, Semantic Mapping

***Abstract*—** This paper describes improving in Semantic Mapping [1], a feature extraction method useful to dimensionality reduction of vectors representing documents of large text collections. This method may be viewed as a specialization of the Random Mapping, method used in WEBSOM project [2]. Semantic Mapping, Random Mapping and Principal Component Analysis (PCA) are applied to categorization of document collections using Self-Organizing Maps (SOM) [3]. Semantic Mapping generated document representation as good as PCA and much better than Random Mapping.

1 Introduction

In self-organization of document collections, high-dimensional data vectors normally represent the documents. The length of these document vectors, in general, equals the number of distinct terms in the vocabulary of the corpus. Thus to turn computational feasible the use of machine learning algorithms the dimensionality of vectors that represent the content of documents, called document vectors, must be reduced to few hundreds, turning essential the use of dimensionality reduction methods.

In the especial case of document organization using SOM, this problem has been addressed by WEBSOM project [2]. The main goal of the WEBSOM was to scale up the SOM algorithm to be able to deal with large amounts of high-dimensional data, thus allowing the construction of document maps of large document collections. This goal was reached due to implementation of a dimensionality reduction method called Random Mapping, shortcuts in the map training, and multi-stage training (large maps initialized from trained small ones). In a number of studies on different text collections the WEBSOM method has been shown to be robust for organizing large and varied collections onto meaningfully ordered document maps. Recently an application of the WEBSOM map of the texts of Encyclopedia Britannica was described in [4].

In particular, the quality of the document maps receive great influence of the document representation and dimensionality reduction methods, given that if the semantic similarity of the documents is clearly expressed by the similarity of the document vectors, then best quality document maps are generated. Motivated by this aspect we proposed in [1] a feature extraction method called Semantic Mapping.

Semantic Mapping has given superior performance than Random Mapping and performance close to Principal Component Analysis (PCA) in text categorization of the K1 Collection [5].

The objective of this paper is to present new methods to generate the projection matrix and report a deep analyses on how Semantic Mapping behave with use of these different methods, extending the work done in [1].

This paper is organized as follows. In Section 2 we review the Random Mapping method. In Section 3 we present the Semantic Mapping method and proposed new methods to construct the projection matrix. Section 4 discusses the methodology and results of the experiments. The classification error in text categorization was used to measure the performance of the dimensionality reduction methods. Section 5 contains the conclusions and future works.

2 Random Mapping

The Random Mapping method (RM) has originally been introduced in [6], and adapted to WEBSOM project [2]. RM is a method generally applicable that approximately preserves the mutual similarities between the data vectors. It consists in constructing a random matrix R and multiplies each document vector by this matrix R.

The matrix R has d rows and n columns. R multiplies each original n-dimensional data vector, denoted by x_j, generating the y_j d-dimensional representation of each one, i.e. the mapping is done taking

$$y_j = R\, x_j.$$

R may be constructed taking random normally distributed values and after normalizing the column vectors to unity Euclidean length, or as a sparse matrix, where a fixed number of ones is randomly put in each column (determining in which extracted features each original feature will participate), and the others elements remained equal to zero. The last method was chosen as default in WEBSOM project after experiments.

Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM 2007)
Published by the Neuroinformatics Group, Bielefeld University, Germany, ISBN 978-3-00-022473-7
All contributions to WSOM 2007 are available online at: http://biecoll.ub.uni-bielefeld.de

The performance of RM was found directly proportional to the number of ones in each column (better performance was generated with 5 ones).
The computational complexity of forming the random matrix is O(*nd*), here *n* and *d* are the dimensionalities before and after the Random Mapping, respectively.

3 Semantic Mapping

The Semantic Mapping (SM) was inspired by Random Mapping method, and also consists in constructing a matrix of projection. Different from Random Mapping, SM incorporates semantics of the original features or dimensions, captured from data-driven form, in construction of new extracted features or dimensions. This method consists of the steps listed in Figure 1.

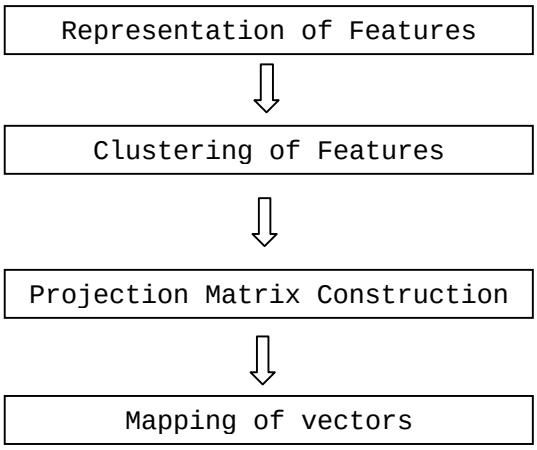

Figure 1: Semantic Mapping method.

Initially, given a matrix P of patterns by features, each original feature or dimension must be represented by a vector, so that the similarity of the vectors approximates the semantic proximity of the original features, i.e. express how correlated they are. A good vector representation of original feature may be done taking the column-vector of a sub-matrix of P with *N* randomly selected patterns. In text categorization, a pattern is a document, a original feature is a term or word, and representing a term by a vector of document frequencies is intuitive given that: the semantics or means of a term can be deduced analyzing the context where this is applied, i.e., the set of documents where it occurs; and co-occurring terms are semantically correlated.
In the second step, feature vectors are grouped in semantic clusters using a clustering algorithm. As similar vectors represent co-occurring original features, clusters of co-occurring features are formed. In text categorization, these clusters typically correspond to topics or subjects present in documents and probably contain semantic related terms. We assign the cluster concept to a desired extracted feature in reduced dimension, thus the number of clusters must be equals the number of extracted features wanted. The clustering algorithm must have linear time complexity to allow SM to be used in large collections.
In previous works, we use only SOM to cluster the features, in this work we also propose and test application of K-means and Leader [10], more fast clustering algorithms. The use of K-means and Leader clustering algorithms to feature clustering in Semantic Mapping is a new contribution, given that SM was originally proposed in [1] using SOM as clustering algorithm.
The construction of the matrix of projection, the third step, is done as follows: each original feature vector is mapped in a fixed number of k clusters that better represent it, i.e., the k first clusters that has the closest codebook to the feature vector. Let *n* be the number of original features and *d* be the number of extracted features, the matrix of projection *M* must be constructed with *d* lines and *n* columns, with m_{ij} equals to one if the original feature *j* was mapped into cluster *i*, zero otherwise. The feature *j* may be mapped into k-best matching clusters if is desired k-ones in each column of *M*. The position of the ones in the columns of the projection matrix indicates which extracted features each original feature will participate. While in RM the position of the ones in each column of R is determined randomly, in SM the position of the ones in each column of M is determined in accordance with the semantic clusters where each original feature was mapped.
The set of matrices of projection generated by SM is a subset of that generated by RM, thus SM also approximately preserves the mutual similarities between the data vectors after projection to reduced dimension.
Finally, the mapping or projection of *n*-dimensional vector representation of a pattern (x_j) to reduced *d*-dimensional vector representation (y_j) is done multiplying the matrix of projection *M* by it ($y_j = M\ x_j$.).
After the mapping, the generated vectors may be optionally normalized in unitary vectors.
The computational complexity of the SM method is O(*ndN*) that is the complexity of the clustering algorithm to generate *d* clusters (number of projected features) from *n* original feature vectors with *N* dimensions (number of document vectors in training set). This complexity is smaller than the complexity of PCA, and still linear to the number of characteristics in the original space as the RM.
The extracted features by SM are, analytically and experimentally [1], more representative of the content of the documents, beyond better interpretable that those generated by RM, allowing generation of best quality SOM maps.
This better performance of SM is also confirmed empirically in the experiments related in the next session.

3.1 Clustering algorithms

We describe here K-means and Leader, algorithms used to feature clustering in SM.

We choose the Leader algorithm because it is the simplest clustering algorithm, and the K-means algorithm because is a well know and efficient algorithm. Both algorithms have linear time complexity in the size of the training set; additionally they are faster than SOM algorithm.
The Leader algorithm [10] is a very fast method for clustering data, the simplest in terms of training time. It requires one pass through the data to put each input pattern in a particular cluster or group of patterns. On the other hand, the algorithm is not invariant to presentation order of the data patterns. Associated with each cluster is a “Leader”, which is one pattern against which new patterns will be compared to determine whether the new pattern belongs to this particular cluster.
Essentially, the Leader algorithm starts off with zero prototypes and adds a prototype whenever none of the existing prototypes is close enough to a current input pattern. The newly created prototype is an exact copy of the current input pattern and is called “Leader” of the new cluster. The cosine of the angle between the input vector and each prototype is used as similarity measure. An influence threshold, whose value ranges from 0 to 1, is a parameter of the system and determines how similar the best matching prototype should be for it to be considered "close enough". In cases when some existing prototype is sufficiently close to the current input pattern, the input pattern is placed in that cluster more close to the pattern. We limited the number of clusters to a maximum value.
K-means algorithm [10] is the most popular clustering algorithm, the reasons behind the popularity are: it is easy to implement; its linear time complexity in the size of the training set; it is order-independent - for a given initial seed set of cluster centers, it generates the same partition of the data irrespective of the order in which the patterns are presented to the algorithm.
We employ a variation of K-means algorithm using cosine similarity measure (cosine of the angle between two vectors). The K-means is an iterative algorithm to minimize a dissimilarity criterion function.
In K-means each cluster is represented by its center, i.e. the mean of all input patterns mapped in it. The centers may be initialized with a random selection of k patterns. Each input pattern is labeled to the cluster of the nearest or most similar center. Subsequent re-computing of the mean for each cluster and re-assigning the patterns to the clusters is iterated until convergence to a fixed labeling or not sufficient improvement after a number iterations or epochs.
A major problem with this algorithm is that it is sensitive to the selection of the initial partition (sensitive to initial seed selection) and may converge to a local minimum of the criterion function value if the initial partition is not properly chosen. In addition, the K-means algorithm, even in the best case, it can produce only hyperspherical clusters.

4 Experiments

In this session is presented the adopted methodology and the results of the experiments. The experiments consist of the application of Semantic Mapping (SM), Random Mapping (RM) and principal component analysis (PCA) to a problem of text categorization using SOM maps as classifier. The goals are: to analyze the behavior of SM with clustering algorithms K-means, Leader and SOM; to compare and analyses the performance of SM, RM and PCA.
Classification error in Text Categorization was used as indicator of quality of the document representation generated by each method, i.e. the performance of each dimensionality reduction method. The classification error was evaluated in the same training and test sets of the K1 collection used in [5].
Document maps with minimal classification error are desired and considered of superior quality. The maps are desired because they represent the document similarity in a close way to the human being. The classification error in a test set is the best measure of the generalization of the cluster structure found by SOM, and can express better the quality of the document map.
Thus, in controlled experiments, the classification error in test set generated by document maps may be used as indicator of quality of the document representation generated by dimensionality reduction method and used in training of SOM map.
The performances achieved by SRM, SM and PCA are compared in projection of *tfidf* document vectors. In *tfidf* representation [7], the documents are represented by real vectors in which each component corresponds to the frequency of occurrence of a particular term in the document (*tf*) weighted by a function of the inverse document frequency (*idf*).

4.1 Preprocessing

The documents categorized belong to K1 collection [5]. This collection consists of 2340 Web pages classified in one of 20 news categories at Yahoo: Health, Business, Sports, Politics, Technology and 15 subclasses of Entertainment (without subcategory, art, cable, culture, film, industry, media, multimedia, music, online, people, review, stage, television, variety).
The document vectors of the collection were constructed using the vector space model with term frequency. These vectors were preprocessed eliminating generic and non-informative terms [5]; the final dimension of the vectors was equal to 2903 terms.
After preprocessing, the document vectors were divided randomly for each category in half for training set and half for test set; each set with 1170 document vectors.
The *tfidf* document vectors representation was calculated as function of term-frequency document vectors as described in [7].

The categories were codified and associated to document vectors as labels.

4.2 Methodology

The performance of the projection methods for *tfidf* document representation was measured, in each dimension of projection (100, 200, 300, 400 and 500), by the mean classification error generated by a SOM map in the categorization of projected document vectors of the test set, trained with the respective projected document vectors of the training set.
The classification error for a SOM map was measured as the percentage of documents incorrectly classified when each map unit is labeled according to the category of the document vectors in training set that dominated the node. Each document is mapped to the map node with the closest model vector in terms of cosine distance. The document vectors of the test set received the category assigned to the node where they were mapped. These SOM maps are denominated document maps.
To measure the performance of the methods SM and RM in relation to the number of ones in each column, for each pair combining dimension and number of ones, were generated 15 matrices of projection for each method. The number of ones in each column in the projection matrix was: 1, 2, 3 and 5.
The PCA method involves the use of Singular Value Decomposition method (SVD) [8] in the extraction of the principal components of the matrix of correlation of the terms in the training set. The correlation matrix was calculated on a *tfidf* matrix of terms by documents. The components are ordered in such way that the first ones describe most of the variability of the data. Thus, the last components can be discarded. Given that the components were extracted, a matrix of projection was constructed for each reduced dimension.
The matrices of projection generated by the three methods had been applied on *tfidf* document vectors, thus forming the projected vectors in the reduced dimensions. The projected vectors of the training and test sets were used to construct the document maps and to evaluate the performance of methods respectively.
The algorithm used for training SOM maps was batch-map SOM [2] because it is quick and have few adjustable parameters.
The SOM maps used in the experiments to cluster vector terms and projected document vectors had a rectangular structure with a hexagonal neighborhood. The Gaussian neighborhood function was used as the neighborhood function. For each topology, the initial neighborhood size was equals to half of the number of nodes with the largest dimension plus one in rough phase and equals to one in fine-tuning phase. The final neighborhood size was always 1 in both phases. The number of epochs of training was 10 in rough phase and 20 in the fine-tuning phase. The number of epochs determines how mild the decrease of neighborhood size will be, since it is linearly decreasing with the number of epochs. The dimensions of document maps were 12x10 units (as suggested in WEBSOM project [9]) with the model vectors with 100, 200, 300, 400 and 500 features. Assuming that there is not prior knowledge in term clustering, the SOM maps had the most squared possible topologies: 10x10, 20x10, 20x15, 20x20 and 25x20, with the model vectors with 1170 features. For each SOM topology maps, randomly initialized configurations with values in (0, 1) interval were used for training.
The parameters of Leader algorithm were influence threshold equals to 0.70, and the number of desired clusters.
The parameters of K-means algorithm were the maximum number of epoch equals to 20, the minimum delta improvement equals to 0.01%, and the number of desired clusters.
Leader, K-means and SOM algorithms use the cosine similarity measure.

4.3 Results

The first step was the evaluation of the number of ones needed in each column of the matrices of projection generated by RM and SM in order to minimize the mean classification errors in the test set. The statistical t-test [11] was used to compare the performances of the methods with different numbers of ones in different reduced dimensions. The t-test was applied on the average and the standard deviation of the classification errors in test set achieved by each method in 15 runs. We observe that 5 ones in each column of the projection matrix improve the performance of RM and SM. For RM, this fact is also related in [2].
Table 1 and Table 2 show the performance of the dimensionality reduction methods in each reduced dimension. The classification error in training and test set, and total time is reported. Total time is the elapsed time in seconds for matrix projection constructing, projection of document vectors, plus the training time of the SOM map with the projected document vectors.
Table 1 shows the performance for PCA. The time required is three or two orders of magnitude bigger than the required for SM, and the performance is not significantly superior to SM using K-means (see Table 2).

Table 1: Experiment Results for PCA

Dim	Training set	Test set	Total Time
100	32,99	41,54	1006,00
200	33,25	37,52	1008,00
300	34,27	39,57	1009,00
400	33,50	40,51	1010,00
500	33,50	40,51	1012,00

The Table 2 shows the performance results for SM and RM methods. SM-K, SM-L and SM-S means Semantic Mapping using K-means, Leader and SOM respectively.

Table 2: Experiment Results for SM and RM

Method	Dim	Training set		Test set		Total Time	
		Mean	stddev	Mean	stddev	Mean	stddev
SM-K	100	32,08	1,18	39,35	1,19	10,66	0,95
SM-L	100	35,58	1,38	42,43	1,78	8,21	0,71
SM-S	100	35,10	0,89	42,76	1,30	15,33	0,97
RM	100	60,79	1,25	73,62	2,02	6,52	0,57
SM-K	200	31,7	1,31	38,93	1,76	13,62	0,92
SM-S	200	34,19	1,01	40,79	0,96	24,34	1,44
SM-L	200	34,15	1,26	40,95	1,62	10,91	0,98
RM	200	56,77	1,55	67,74	1,70	8,28	0,75
SM-K	300	31,98	1,15	38,55	1,37	16,42	1,25
SM-L	300	33,58	1,48	39,95	2,20	12,49	1,16
SM-S	300	34,61	1,21	41,20	1,51	31,73	2,17
RM	300	54,05	2,09	64,88	2,98	9,75	0,78
SM-K	400	31,97	1,11	39,25	1,48	20,06	1,41
SM-L	400	32,80	1,54	39,74	1,60	14,67	1,00
SM-S	400	34,03	1,13	40,88	1,75	40,84	2,39
RM	400	50,56	1,77	59,3	2,63	10,89	0,71
SM-K	500	31,73	1,24	38,73	1,56	22,31	1,93
SM-L	500	32,16	1,24	39,13	1,74	16,88	1,50
SM-S	500	33,64	1,34	40,00	1,55	50,97	5,20
RM	500	50,31	2,57	59,13	3,14	12,04	0,85

Figure 2 shows a graphical representation of the experiments results. The graph shows the classification error in test set as function of reduced dimension of document vectors for RM, SM and PCA methods. For SM and RM are plotted the mean classification error and the bars denote one standard deviation over 15 runs.

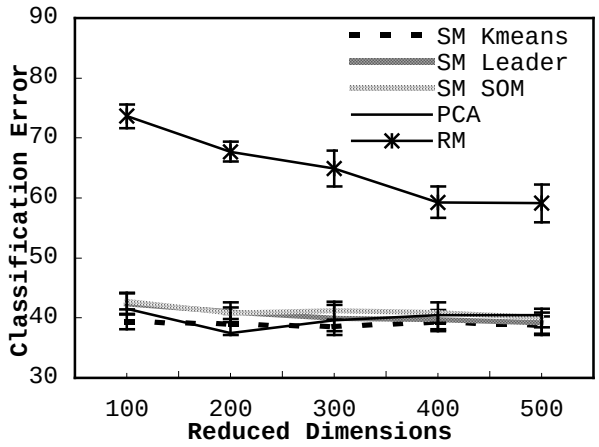

Figure 2: Classification error as function of reduced dimension for RM, SM and PCA

Figure 2 shows that SM performance is very close to PCA performance, and that SM is better than RM. The mean classification error of RM is superior to 59% for all the reduced dimensions. The RM classification error decreases significantly with increasing of the dimension of projection, as pointed out in [2]. In relation to SM this fact is also true when using SOM or Leader, but using K-means the performance is practically stable.

Figure 3 shows the graph of Figure 2 with only SM and PCA performance plotted. It shows that SM performance is very close to PCA performance, and that only for one reduced dimension (200) the mean performance of the SM using K-means is lower than the performance of PCA. For PCA method, for reduced dimensions bigger than 200, the classification error increases with the increasing of the dimension of projection, this is because the principal components after 200 incorporate the variability of the noise.

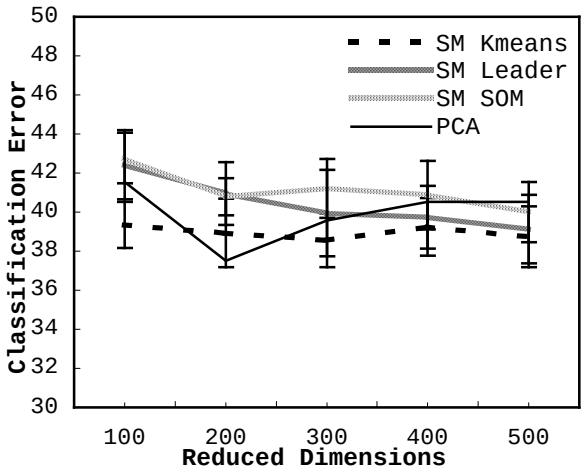

Figure 3: Classification error as function of reduced dimension for SM and RM.

In Table 3 we use t-test to compare the performance of the methods RM and SM. The numbers 1, 2, 3 and 4 denotes the methods SM-K, SM-L, SM-S and RM respectively. The following range codification of the P-value of the t-test was used [12]: “>>” and “<<” mean that the P-value is lesser than or equal to 0.01, indicating a strong evidence of that a system generates a greater or smaller classification error than another one respectively; “<” and “>” mean that the P-value is bigger than 0.01 and minor or equal to 0.05, indicating a weak evidence that a system generates a greater or smaller classification error than another one respectively;"~" means that the P-value is greater than 0.05 indicating that it does not have significant difference in the performance of the systems.

Table 3: Comparison of SM and RM Performance

Dim\Method	2-1	3-1	3-2	4-1	4-2	4-3
100	>>	>>	~	>>	>>	>>
200	>>	>>	~	>>	>>	>>
300	>	>>	>	>>	>>	>>
400	~	>>	>	>>	>>	>>
500	~	>	~	>>	>>	>>

Table 3 shows that RM had the worst performance. SM had best performance using K-means algorithm, but SM

Proceedings of the 6th International Workshop on Self-Organizing Maps (WSOM 2007)
Published by the Neuroinformatics Group, Bielefeld University, Germany, ISBN 978-3-00-022473-7
All contributions to WSOM 2007 are available online at: http://biecoll.ub.uni-bielefeld.de

using any cluster algorithm had better performance than RM. SM using Leader is better than SM using SOM because it had the same performance or superior performance than SM using SOM, also it had the same performance than SM using K-means in some dimensions. Thus, the use of K-means or Leader in SM is a better choice than SOM because those methods had the same or better performance than SOM and smaller total time to generate the document map.

5 Conclusions

Analytically and experimentally, the features extracted by Semantic Mapping showed to be more representative of the content of the documents and better interpretable than those obtained through Random Mapping.

SM showed to be a viable alternative to PCA in the dimensionality reduction of high-dimensional data due to the same or better performance than PCA and the computational cost linear to the number of features in the original space, as RM.

SM had better performance using the clustering algorithms K-means and Leader than using SOM. The best performance of SM is obtained using K-means as clustering algorithm.

Future investigations should consider testing SM, RM and PCA methods to dimensionality reduction of others document collections; testing the influence of the number of documents in SM performance; and to elaborate and to evaluate new methods to cluster features and to construct projection matrices in SM.

Acknowledgements

The authors would like to thank CNPq and CAPES (Brazilian research agencies) for their financial support.

References

[1] R. F. Correa, and T. B. Ludermir, Dimensionality Reduction by Semantic Mapping, Proceedings of VIIIth Brazilian Symposium on Neural Networks, 2004.

[2] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero and A. Saarela, Self Organization of a Massive Document Collection, *IEEE Transaction on Neural Networks*, Vol. 11, n. 3, May 2000, pp. 574-585.

[3] T. Kohonen, Self-Organized formation of topologically correct feature maps, *Biological Cybernetics*, Vol.43, pp.59-69, 1982.

[4] K. Lagus, S. Kaski, and T. Kohonen, Mining massive document collections by the WEBSOM method *Information Sciences*, Vol 163, No. 1-3, pp. 135-156, 2004

[5] D. Boley, M. Gini, R. Gross, E. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher and J. Moore, Partitioning-based clustering for web document categorization, *Decision Support Systems*, Vol .27, 1999, pp. 329-341.

[6] H. Ritter, and T. Kohonen, Self-organizing semantic maps, *Biological Cybernetics*, Vol. 61, 1989, pp.241-254.

[7] F. Sebastiani, Machine Learning in Automated Text Categorization, Proc. *ACM Computing Surveys*, Vol. 34, No. 1, March 2002. pp. 1-47.

[8] G. E. Forsythe, M. A. Malcolm, and C. B. Moler, Computer Methods for Mathematical Computations, Prentice Hall, 1977.

[9] S. Kaski, Dimensionality reduction by random mapping: Fast similarity computation for clustering, Proc. IJCNN'98 Int. Joint Conf. Neural Networks, vol. 1, 1998, pp.413-418.

[10] J. A. Hartigan, Clustering Algorithms, John Wiley & Sons,Inc., New York, USA, 1975.

[11] I. H. Witten and E. Frank, Data Mining - Pratical Machine Learning Tools and Techniques with Java Implementations. San Francisco: Morgan Kaufmann Publishers, 2000.

[12] Y. Yang, X. Liu, A re-examination of text categorization methods, Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99), 1999, pp. 42–49.