51

REDUCTION TECHNIQUES FOR INSTANCE BASED TEXT CATEGORIZATION

Peter Bednár, Tomáš Futej
Dept. of Cybernetics and Artificial intelligence, Technical University of Kosice, Letna 9, 042 00 Košice, SLOVAKIA

One of the most common problems in instance-based learning of text categorization is high dimensionality of feature space and problem of deciding which instances to store for use during generalization. These problems can be solved with use of reduction methods. In this paper, comparison of three reduction techniques for feature space reduction and one algorithm for reduction of storage requirements is presented. These techniques were combined with k-NN (k-Nearest Neighbors) classifier, which is one of the top-performing methods in the text classification tasks. We describe the benefit of this combination of methods and present results with the Reuters-21578 dataset.

1. INTRODUCTION

Text categorization is the problem of automatically assigning predefined categories (or classes) to text documents [3]. While more and more textual information is available, effective information retrieval is difficult without indexing of document content [1].

Document categorization is one solution to this problem. One of the top performing methods for text categorization is instance based, k-nearest neighbors, classifier. Main disadvantage of this method is high time complexity and high memory requirements. In this paper, we describe various reduction techniques, which can solve these problems.

2. INSTANCE BASED LEARNING - kNN CLASSIFIER

Example-based classifiers do not build an explicit, declarative representation of the categories, but rely on the category label attached to the training documents similar to the test document.

The kNN algorithm [3, 4, 5] is simple: given a new document, the system finds the k nearest neighbors among the training documents, and uses the categories of the

neighbors to weight the category candidates. The similarity score of each neighbor document to the new document is used as the weight of the categories of the given neighbor. By sorting and thresholding the scores of candidate categories, binary category assignments are obtained. For convenience, the cosine value of two document vectors is used to measure the similarity between the documents, although other similarity measures are possible.

3. FEATURE SELECTION

One difficulty of text categorization problems is high dimensionality of the feature space. Feature space can consist of hundreds or thousands of unique terms (words or phrases) that occur in documents. We have evaluated combination of three methods, including document frequency, information gain and mutual information.

3.1 Document frequency thresholding

Document thresholding (DF) [2] is one of the simplest techniques for feature space reduction. Document frequency is the number of documents in which a term occurs. All unique terms that have document frequency in training set less than some predefined threshold were removed. The basic assumption is that rare terms are either non-informative for category prediction, or not influential in global performance. Improvement in categorization accuracy is also possible if rare terms happen to be noise terms.

3.2 Information gain

Information gain (IG) [2] measures the number of bits of information obtained for category prediction by knowing the presence of a term in a document. The information gain of term t is defined to be:

$$G(t) = -\sum_{i=1}^{|C|} P(c_i)\log P(c_i) + P(t)\sum_{i=1}^{|C|} P(c_i \mid t)\log P(c_i \mid t) + P(\bar{t})\sum_{i=1}^{|C|} P(c_i \mid \bar{t})\log P(c_i \mid \bar{t}) \quad (1)$$

where $P(c)$ is the probability of the category c, and $P(c \mid t), P(c \mid \bar{t})$ denotes conditional probability of category c given the presence or absence of term t.

3.3 Mutual information

Mutual information (MI) [2] is criterion commonly used in statistical language modeling of word associations. Mutual information can be estimated using:

$$I(t,c) \approx \log \frac{AN}{(A+C)(A+B)} \tag{2}$$

where A, B, C, and D are cells of the two way contingency table of term t and category c and $N = A + B + C + D$.

Given a training corpus, for each unique term we have computed the information gain or mutual information and removed from the feature space those terms whose IG or MI was less than some predetermined threshold.

4. INSTANCE REDUCTION

For instance selection, we have adopted algorithm called Decremental Reduction Optimalization Procudere 4 (DROP) [6]. This procedure is decremental, meaning that it begins with the entire training set, and then removes instances that are deemed unnecessary. DROP4 uses following basic rule to decide if it safe to remove an instance i from the set of training instances S:

Remove instance i from S if at least as many of its associates in T would be classified correctly without i.

To see if an instance i can be removed using this rule, each *associate* (i.e. each instance that has i as one of its neighbors) is checked to see what effect the removal of i would have on it. Instance will be removed if its removal does not hurt the classification of the instances in T (according to the F1 accuracy measure). DROP4 removes instances in the center of a category cluster and it can remove noisy instances, because a noisy instance i usually has associates that are mostly of a different category. DROP4 initially sorts the instances by the distance to their nearest enemy (i.e. nearest instance with a different category), because order of removal can have influence on instance reduction. DROP4 algorithm has additional noise-filtering pass before sorting of instances, which is based on rule similar to *Edited Nearest Neighbor* rule. It states that any instance misclassified by its k nearest neighbors is removed (if it does not hurt the classification of its associates).

5. EXPERIMENTS

We have tested described reduction techniques and their combinations on Reuters-21578 corpus. We use the ModApte version, which was obtained by eliminating unlabelled documents, and selecting the categories, which have at least one document in the training set and the test set. This process resulted in 90 categories in both the training and testing set.

The first experiment was used to setup basic parameters (such as k and c – which is the threshold for category assigning). The accuracy of the baseline k-NN classifier (defined as F1 measure) was 0.77. We can probably achieve the better result with

different shareholding techniques, for example, sets threshold for each category by using cross-validation) [4].

The second experiment was oriented on selection of relevant terms based on document frequency of unique terms. According to the results for document frequency reduction (Figure. 1), useful terms have DF between 50 to 2000. Classification with selected terms does not decrease the performance of the classifier and feature space was reduced to 5.6% of the original size. The similar results are for information gain reduction (Figure. 2). If we have removed terms with lower value of IG under the 90% of terms, precision increase to 0.78. Time needed for classification was reduced 5 times. MI thresholding has different effect on performance on k-NN classifier (Figure. 3). MI is biased towards low frequency terms and this make significant accuracy loss in text categorization.

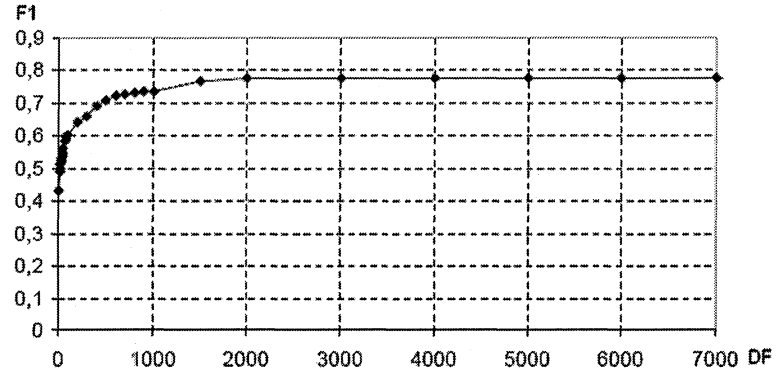

Figure 1 – Document frequency thresholding feature space reduction

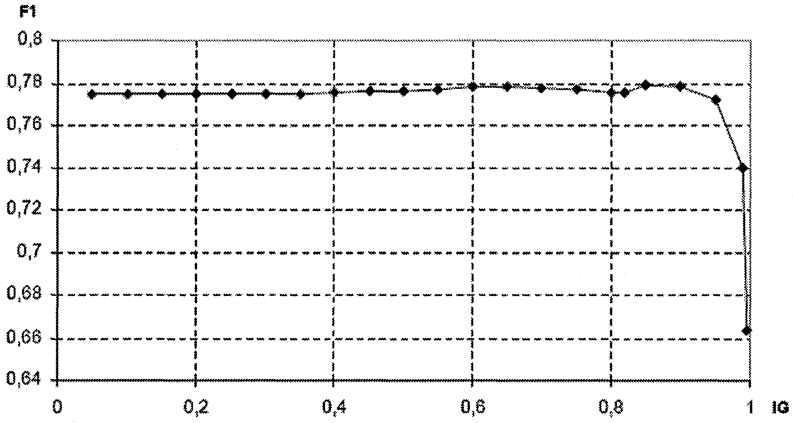

Figure 2 – Information gain feature space reduction

According to the results of experiments in (Figure. 4), the performance of classification was lower when we have used DROP4 algorithm (number of instances

was reduced to 36% of original size of training set). Highest influence on this has first phase of DROP4 that is targeted to remove noisy instances. At this point, performance goes down by 2% with remove of 101 instances. Since we have categories, which have only few instances, these instances are consider being a noise and are removed by first phase of the algorithm. When we have used only basic instance of DROP algorithm called DROP1 without ENN phase the performance of classifier has risen to 0.79.

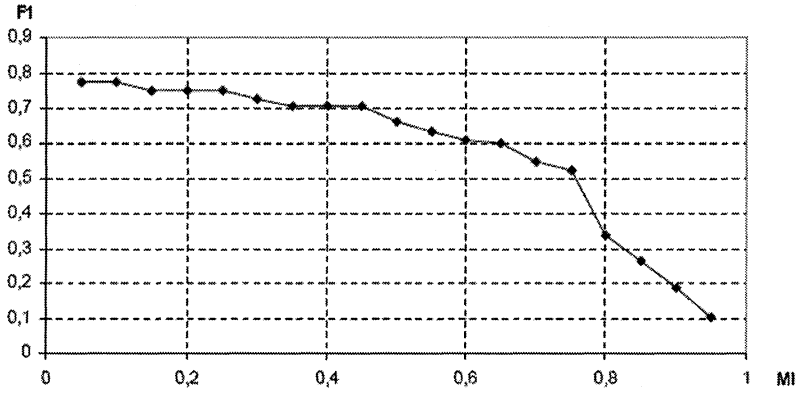

Figure 3 – Mutual information feature space reduction

6. CONCLUSION

This paper describes various algorithms for feature space and instance space reduction. In the first section of the paper, we have described some of the problems of instance based learning in text categorization task and how these problems can be solved by reduction techniques. In second section, we have introduced the results of the practical test of reduction techniques on standard Reuters benchmark. According to the results, the reduction techniques have positive influence on instance-based classification. We have achieved better performance and lower time and space complexity with DF and IG thresholding for feature space reduction and modified DROP algorithm for instance space reduction.

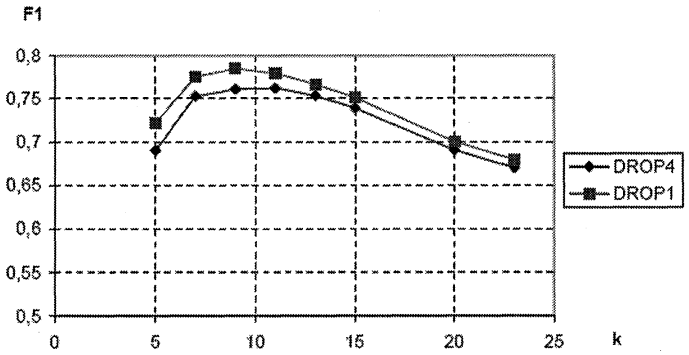

Figure 4 – F1 measure of instance space reduction techniques

6.1 Acknowledgement

This work is done within the VEGA project 1/1060/04 "Document classification and annotation for the Semantic web" of Scientific Grant Agency of Ministry of Education of the Slovak Republic.

7. REFERENCES

1. Ontology-based Information Retrieval, by J.Paralic and I.Kostial. In Proc. of the 14th International Conference on Information and Intelligent systems, IIS 2003, Varazdin, Croatia, ISBN 953-6071-22-3, 23 - 28, 2003.
2. Yang, Y., Pedersen J.P. A Comparative Study on Feature Selection in Text Categorization Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97), 1997, pp412-420.
3. Yiming Yang and Xin Liu A re-examination of text categorization methods. Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99, pp 42--49), 1999.
4. Yiming Yang A study on thresholding strategies for text categorization, Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01), pp 137-145, 2001.
5. Han, E. H., Karypis G., Kumar, V. Text Categorization Using Weight Adjusted k-Nearest Neighbours.
6. Wilson D. R., Martinez T., Reduction Techniques for Instance-Based Learning Algorithms, Machine Learning 28(3):257-286, 2000.