Support Vector Machines for Text Categorization Based on Latent Semantic Indexing

Yan Huang

Electrical and Computer Engineering Department

The Johns Hopkins University

huang@clsp.jhu.edu

Abstract

Text Categorization(TC) is an important component in many information organization and information management tasks. Two key issues in TC are feature coding and classifier design. In this paper Text Categorization via Support Vector Machines(SVMs) approach based on Latent Semantic Indexing(LSI) is described. Latent Semantic Indexing[1][2] is a method for selecting informative subspaces of feature spaces with the goal of obtaining a compact representation of document. Support Vector Machines[3] are powerful machine learning systems, which combine remarkable performance with an elegant theoretical framework. The SVMs well fits the Text Categorization task due to the special properties of text itself. Experiments show that the LSI+SVMs frame improves clustering performance by focusing attention of Support Vector Machines onto informative subspaces of the feature spaces.

1. Introduction

As more and more information is available on the internet, there is an ever growing interest in assisting people manage the huge amount of information. Information routing/filtering, identification of objectionable materials or junk mail, structured search/browsing, and topic identification,etc, these are all hot spots in current information management. The assignment of texts to some predefined categories based on their content, namely Text Categorization(TC), is an important component among these tasks. Two key issues in Text Categorization are feature coding and classifier design. A lot of work in both these two aspects have been done.

Feature extraction, which is basically a method of document coding, automatically construct internal representations of documents. The basic principles in document coding are: Firstly, it should be amenable to interpretation by the classifier induction algorithms; Secondly, it should compactly capture the meaning of document and therefore is computationally flexible and feasible[4]. The aggressivity of feature selection is defined as $(1-r'/r)$, in which r is the number of original feature set and r' is the number of reduced optimal feature set. A higher aggresivity results lower computational expense; but meanwhile, it may also curtail the classification performance of the classifier. Therefore, an appropriate coding scheme is a first of all issue in TC. Mutual Information feature selection[5], filtering approach[6], and etc, are effective feature selection methods widely used in TC. Latent Semantic Indexing(LSI) is an altenate coding scheme, which extracts the underlying semantic structure of a corpus by determining the most significant statistical factors in the weighted word space. The advantages lie not only in reducing the dimensionality, but also in digging factors which account for higher-order associations between groups of words that may not be present in individual documents[7].

As for classifier design problem, a number of statistical classification and machine learning techiniques have been applied in TC. These include multivariate regression models[8][9][10], probabiloistic Bayesian models[11], nearest neighbor classfiers[12], decision trees[13], adaptive decision trees[14], neural networks[15][16], symbolic rule learning[17] and Support Vector Machine Learning[18]. Many of them have reported decent results. Due to the following properties of text itself, SVMs well fit TC task: (1) High dimension property. Since SVMs use overfitting protection, they have the protential to handle these large feature spaces. (2) Sparseness property. Document reprentation are turned to be very sparse, with only a few non-zero items, while most others are zeros. It has been proved both theoretically and practically that SVMs are well suited for both dense and sparse problems.[19] (3) Most text categorization problems are linearly separable.[4]

The following part of the paper will introduce the approach of Text Clustering by Sopport Vector Machines based on Document Latent Semantic Indexing(LSI): 2. Latent Semantic Indexing; 3. SVMs method for clustering; 4. Experiments and Analysis; 5. Conclusion.

2. Latent Semantic Indexing

Latent Semantic Indexing adopts a vector model of semantics based on word co-occurrences. The assumption is that words, that tend to occur togetherm or tend to occur with similar words, are considered to be semantically similar. The documents are treated as semantically cohesive set of words, such as paragraphs, articles from newspapers, newsgroup articles, etc. This is also referred to as a bag-of-words model since structure within the documents is not maintained.

To build the LSI model, a matrix representation of training document is created first, with rows corresponding to words in the vocabulary and columns to documents. Each entry in the matrix is a weighted frequency of the corresponding term in the corresponding document. This weighting is to reduce influence of frequently occurring terms, such as the function words. More details can be found in[20]. The next step is to reduce this large sparse matrix into a compressed matrix based on singular value decomposition(1):

$$M = USV \quad (1)$$

The original matrix M is decomposed into a reduced rank term matrix U, a diagonal matrix of singular values S and a document matrix V. The row vector of matrix U and the column vector of matrix V are the projections of word vectors and document vectors into singular value space. Thus words and documents are represented in a much more compact way compared to the original representation. Depending on the different tasks, the number of selected singular value varies. 50 to 400 are typical choices in many regular tasks. Figure1 are examples of some word representations:

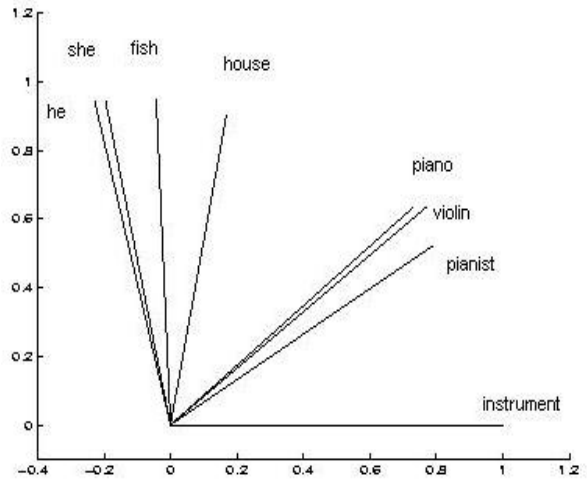

Figure1 Examples of word representation

3. SVMs Method for Clustering

Support vector machines are based on the Structure Risk Minimization principle from computational theory. The basic idea is to find a hypothesis for which we can guarantee the lowest true error. The true error of the hypothesis is the probability that it will make an error on an unseen and randomly selected test example.

Support vector machine method is one solution for data overfitting problem in neural networks. The problem occurs because traditional backpropagation training algorothms are gradient descent algorithms with no mechanism to determine the optimal point at which to stop descending the error gradient.

There are several kinds of SVMs for choice. In this paper, the simples polynomial SVMs are used for clustering.

4. Experiments and Results

4.1 Data Sets

The first data set is the WSJ87,88,89 text data. Randomly select 80,000 documents as the training set. A vocabulary of 20,000 most frequent words is used.

The second data set is the famous Reuters-21578 text categorization collection (it is available for downloading at http://www.research.att.com/~lewis/reuters21578.html). 12,902 stories that had been classified into 118 categories is used in this experiment. Following "ModeApte" splitting, 75% of the data(9603 stories is used as the training data for classifier training; 25%(3299 stories) is used as the testing data. Due to time limitation, 5 most frequent categories, which are shown in table1, are selected as the experiment classifier. These 5 categories consist 60 % of the whole training set.

Category Name	Num Train	Num Test
Earn	2877	1087
Acquisitions	1650	719
Money-fx	538	179
Grain	433	149
Crude	389	189

Table1 Top5 Categories: Number of Training/Test Items

4.2 Experiment

The first data set is used to get the word representation in S-space.(Note: This is due to the time limit of my experiment. The ideal way is to extract word

representation on the same data set as text catigorization data, Reuters-21578 text). 20,000-by-80,000 sparse matrix is generated to represent the WSJ training corpus, with each column being the vector representation of document in the original space. The detail about coding method can be referred in [20]. Then the SVD toolkit[21][22] is used for Singular Value Decomposition. Each word in the vocabulary is represented as a 125-dimension vector in S-space.

To encode Reuters training text, each text is represented as the normalized summation of words appearing in that document.

The Next step is to train a SVM for each top 5 categories described in Table1. Simple polynomial SVMs are used here because they provide good generalization accuracy and are fast to learn. After the learning is finished, 125 feature weights are obtained for each SVM corresponding to each top 5 category. Two parameters of a sigmoid function is learned to transform the binary output of SVM to probability.

In testing phase, encode the test text the same way as described above and fill it into 5 SVMs obtained from training procedure. Rank the output probabilities of the 5 SVMs, choose the biggest one as the classification result.

4.3 Results

The most popular measures for classificaton performance are based on precision and recall. Precision is the proportion of items placed in the catogory that are really in the category, recall is the proportion of items in the category that are actually plcaed in the category. Here the average of precision and recall, namely breakeven point, is used to evaluate the classification results. The average result is measured using micro-avarage scores, which gices equal weight to every documentm. Therefore, it is considered to be a per-document average or an average over all the document / category pairs[23].

Figure3 shows the initial experiments of classification results:

	Linear SVMs
Earn	97.0%
Acq	93.2%
Money-fx	75.0%
Gain	94.0%
Crude	90.4%
Average TOP5	93.3%

Figure 3 Initial Experiment Results

Further experiments include extending to the whole categories set will be done in the future work. The initial experiment on the top 5 categories shows that LSI+SVMs performs well in TC task.

5. Summary

This paper introduces Support Vector Machines for Text Categorization based on Documnet Latent Semantic Indexing. Experiments show that LSI is an effective coding scheme. It captures the underlying content of document in semantic sense. SVMs well fit for text catigorization task due to the properties of text. LSI+SVMs shows to be a promising scheme for TC task. Due to time limit, further experiments on the whole 135 catogories will be done in the future work. Future direction include how to use this scheme to solve the defficient amount of hand-labelled training data problem.

6. Reference

[1] M.W. Berry, S.T. Dumais, and T.A. Letsche, "Computational Methods for Intelligent Information Access", Proceedings of Supercomputing'95, San Diego, CA, December 1995.

[2] Michael W. Berry and Susan T. Dumais, and Gavin W. O'Brien, December 1994. Published in SIAM Review 37:4 (1995), pp. 573-595.

[3] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, Vol. 2, Number 2, p. 121-167, 1998 .

[4] Luigi Galavotti, Fabrizio Sebastiani, Maria Simi, "Feature Selection and Negative Evidence in Automated Text Categorization (2000)

[5] Yang, Y., Pedersen J.P. A, "Comparative Study on Feature Selectioin Text Categorization", Proceedings of the Fourteenth International Conference on Machine Learning (ICML'97), 1997, pp412-420.

[6] H. T. Ng, W. B. Goh, and K.L.Low. "Feature selection, perception learning , and a usability case study for text categorization", Proceedings of SIGIR-97, 67-73, 1997.

[7] Gary Noel Boone, "Extreme Dimensionality Reduction for Text Learning Cluster-generated Feature Space", Ph.D Thesis, Geogia Institute of Technology, August 2000.

[8] Yang, Y. and Chute, C.G. An example-based mapping method for text categorization and retrieval. ACM Transactions on Information Systems, 12(3), 252-277, 1994.

[9] Fuhr, N., Hartmanna, S., Lustig, G., Schwantner, M., and Tzeras, K. Air/X-A rule-based multi-stage

indexing system for large subject fields. In Processings of RIAO'91, 606-623, 1991.

[10] Schutze, H., Hull, D. and Pedersen, J.O. A comparison of classifiers and document representations for the routing problem. In SIGIR Conference in Research and Development in Information Retrieval, 229-237, 1995.

[11] Yang, Y. An evaluation of statistical approaches to text catigorization. CMU Technical Report, CMU-CS-97-127, April 1997.

[12] Lewim D.D. and Ringuuette, M.. A comparoson of two learning algorithms for text categorization. In Thirs Annual Symposium on Document Analysis and Information Retrieval, 81-93, 1994.

[13] Sholom M. Weiss, etc, Maximizing Text-Mining Performance, IEEE Intelligent Systems 2-8, July/August,1999.

[14] Wiener E., Pedersen, J.O. and Weigend, A.S. A neural network approach to topic spotting. In Processings of the Fourth Annual Symposium on Document Analysis and Information Retrieval(SDAIR'95), 1995.

[15] Schutzw, H., Hull, D. and Pedersen, J.O. A comparison of classifiers and document representations for rhe routing problem. In SIGIR, 229-237,1995.

[16] Apte, C., Damerau, F. and Weiss, S. Automated learning of decision rules for text categorization. ACM Transactions on Information System, 12(3), 233-251, 1994.

[17] J.Kiven, M.Warmuth, and P.Auer. The perceptron algorithm vs. window: Linear vs. logarithmic mistake bounds when few input variables are relevant. In Conference on Computeratiobal Learning Theory, 1995

[18] Cohen, W.W. and Singer, Y. Context-sentive learning methods for text categorization in SIGIR 307-315, 1996.

[19] Joachines, T. Text categorization with support vector machines: Learning with many relevant features. In Proceedings 10th Eurospeech Conference on Machine Learning(ECML), Springer Verlag, 1998.

[20] .R. Bellegarda , "A Multi-Span Language Modeling Framework for Large Vocabulary Speech Recognition," IEEE Trans. Speec and Audio, in press.

[21] M.W. Berry, "Large-Scale Sparse Singular Value Computations," Int. J. Supercomp. Appl., Vol. 6, No. 1, pp. 13-49, 1992. Extreme Dimensionality Reduction for Text Learning: Cluster-generated Feature Spaces.

[22] M. W. Berry et al., SVDPACKC: Version 1.0 User's Guide, Tech Rep. CS-93-194, Universoty of Tennessee, 1993.

[23] Yang, Y., "An Evaluation of statistical approaches to text categorization.", Journal of Information Retrieval, 1999, Vol 1, No. 1/2, pp 67--88.