

AN EFFECTIVE DOCUMENT CATEGORY PREDICTION SYSTEM USING SUPPORT VECTOR MACHINES, MANN-WHITNEY TECHNIQUES

P.Sundara Vadivel¹, S.Navaneetha Krishnan², D.Yuvaraj³

¹Associate Professor- Department of Computer Science and Engineering

²Professor- Department of Electronics and Communication Engineering,

^{1,2}SACS MAVMM Engineering College, Madurai, Tamilnadu, India

³Professor- Department of Information Technology,

Karpagam College of Engineering, Coimbatore, Tamilnadu, India

Abstract: Effective document categorization is defined as the number of relevant documents that are joined together with respect to user query. Context of user information need be classified into document and query context. Query context contains query terms, user intention and statistics. Document context deals with frequent co-occurrence matrix of the header and surrounding words. Our document selection techniques include Fisher Criterion, Golub Signal-to-Noise, traditional t-test and Mann-Whitney rank sum statistic. The classification algorithm include support vector machine (SVMs) with several kernels and k-nearest neighbor (k-nn). In this paper, we present a novel document category prediction in IR to enhance the performance of the retrieval system. The best data fusion technique that can be combined with clustering SVMs and Mann-Whitney Rank sum statistic. The study show that our approach is more efficient than traditional approaches

Keywords: Document selection, support vector machines, Mann-Whitney.

1. Introduction

Different techniques have been used to develop result prediction systems. The problem of finding user information need in various documents has become more serious considering increasing number of databases and data warehouse. The goal is to provide a single search interface that provides access to available documents and similarity measures are grouped together for near-efficient and effective information retrieval for category prediction. Classifying and grouping the data based on the information in the document. A good document classifier called Associative Text Classifier [ATC] is implemented in DCPS. The ATC uses the advantage of association rule mining. This step involves applying specialized association rule to identify the patterns in the document. The results of the document classifier cannot simply be accepted, but must be carefully evaluated and interpreted. The association rule mining

process is an iterative process. After the initial run of the mining process is complete, the user will evaluate the results and decide whether further work is necessary or if the results are adequate. It is fast enough in training the system for a particular category by set of related documents and in categorizing a document towards a trained category. The system can be read, understood and modified by humans. The effectiveness of the system can be increased manually by tuning the classification rules. Classifiers are of two types namely Single Class Classification [SCC] and Multiple Class Classification [MCC]. In SCC each document is assigned a unique class label. In MCC a document could be classified in many classes simultaneously.

The rest of the document is organized as follows. Section 2 discusses the literature survey. Section 3 consists of brief description of categorization model and explains the Associative Text Classifier construction and system architecture for category prediction. Section 4 consists of a Support Vector Machines and Mann-Whitney Rank sum statistic Chapter 5 consists of the Discussion.

2. Related Work

2.1 Guttman's point alienation

The pair wise similarity measure include: The number of documents in the intersection of the two lists of returned documents (I)

The correlation coefficient from a linear regression of the scores of documents in the intersection of the two systems (C), which is actually the r^2 value of a regression which uses one system's scores to predict the other's.

The number of relevant documents returned by one system but not the other divided by the total number of relevant documents returned by that system

$$GPA = \frac{\sum_{x,x'} (\rho_1(x,q) - \rho_1(x',q)) (\rho_2(x,q) - \rho_2(x',q))}{\sum_{x,x'} |\rho_1(x,q) - \rho_1(x',q)| |\rho_2(x,q) - \rho_2(x',q)|}$$

2.2 Lees overlap measure

Lee's [Lee, 1997] overlap measures, O_{rel} and O_{nonrel} , which measure the proportion of relevant and nonrelevant documents in the intersection of the two lists. These two measures are calculated as:

$$O_{rel} = \frac{2 * I_{rel}}{R_1 + R_2}$$

$$O_{nonrel} = \frac{2 * I_{nonrel}}{N_1 + N_2}$$

where R_i is the number of relevant documents and N_i is the number of non relevant documents returned by the system i respectively. The ratio of the two systems found to be an important predictive factor for the improvement of the combination. The similarity measure is the two systems on relevant document is less important than on relevant ones. After normalizing the scores for each system on each query by dividing their respective means we found the optimal combination for each possible. For each feature, we use one of the statistical methods such as the traditional t-test. Large score suggests that the corresponding feature has different expression levels in the relevant and irrelevant documents and thus is an important feature and will be selected for further analysis. Besides that some researchers used a variation of correlation coefficient to select features, for example Fisher Criterion [13] and Golub Signal-to-Noise.

3. Document categorization

Text categorization has a long history, starting in the early 1960s. Rocchio's algorithm [3] is the classical method in information retrieval, being used in routing and filtering documents. Classifiers based on probabilistic models have been proposed starting with the Maron's algorithm in 1962 and continuing with Naïve Bayes [5] that proved to perform well. ID3 and C4.5 are well known packages whose cores are making use of decision tree to build automatic classifiers [2,4] K-nearest neighbor (K-NN) is another technique used in text categorization [7]. Another method to construct a text categorization system is by an inductive rule learning method. This type of classifier is represented by a set of rules in disjunctive normal form that best cover the training set [3, 7]. The use of bigrams was improved text categorization accuracy as opposed to unigrams use [6]. In the latest decade neural networks and Support Vector Machines used in text categorization.

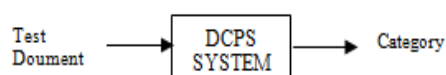


Figure 3.1. General block Diagram of DCPS system

3.1 Phases of Document Categorization

In this paper we present a method to build a categorization system that merges association rule mining task utilizing a priori algorithm. Given a set of training samples of the transaction database, a number of steps are followed until the corresponding data in the document is found. The set of text related to a given text is mostly composed of synonyms and near-synonyms. In addition to these, relationships can be induced by patterns of co-occurrence within documents. Such relationships are usually of a hierarchical nature and most often indicate broader or narrower related terms.

The Document categorizer consists of two major phases namely, training and testing phase.

Training phase

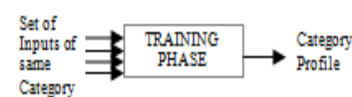


Figure 3.1. Training Phase

As in fig 3.1 the system gets trained by analyzing set of input documents of same category. In the training phase, some huge collection of documents of the same category is given to the system to train the system for that particular category. Like wise the system has to get trained for different categories.

Testing Phase

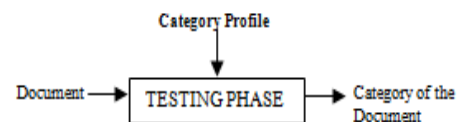


Figure 3.2. Testing Phase

After training the system, it can be used to identify the category of a particular document by giving a single document as input to the testing phase as shown in fig. 3.2. The testing phase does the feature extraction for this document and these features are compared with category profiles that are prepared in the training phase. For the satisfactory outcome of the comparisons, the testing phase can predict the category that it belongs to.

3.2 Detailed Architecture of Categorizer

Due to historical reasons, documents in a collection are frequently represented through a set of index terms or keywords. Keywords provide a logical view of the document. The retrieval system adopts a full text logical representation of the documents and recognizes the internal structure normally present in a document (e.g.,

chapters, sections, subsections, etc.). For centuries, indexes were created manually as categorization hierarchies. In fact, most libraries still use some form of categorical hierarchy to classify their volumes (or documents). In the system-centered perspective, the information retrieval problem consists mainly of building up efficient indexes, processing user queries with high performance, and developing ranking algorithms which improve the 'quality' of the answer set. According to this view, keyword based query processing might be seen as a strategy which is unlikely to yield a good solution to the information retrieval problem in the long run.

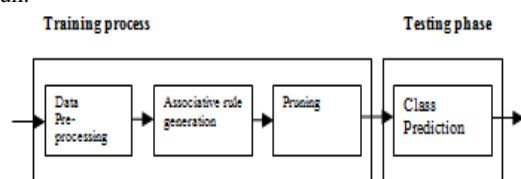


Figure 3.3. Training Phase

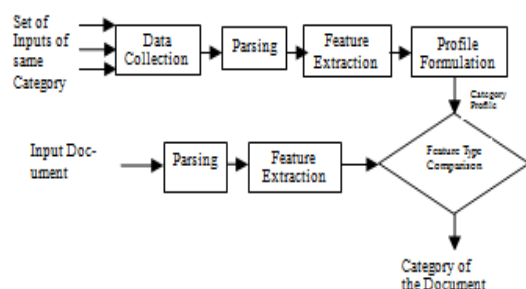


Figure 3.4. Detailed Architecture of DCPS

Let, document D_i is assigned to a set of categories $C = \{c_1, c_2, c_3, \dots, c_m\}$. After word pruning the set of index terms $T = \{t_1, t_2, t_3, \dots, t_n\}$ is retained, the following transaction is used to model the document, D_i : $\{c_1, c_2, c_3, \dots, c_m, t_1, t_2, t_3, \dots, t_n\}$ and the association rules are discovered from such transactions representing all documents in the collection. The Figure 3.4 explains the detailed architecture of document category prediction system.

3.2.1 Document pre-processing

Document pre-processing is a procedure which can be divided mainly into five text operations:

1. Lexical analysis of the text to identify the digits, punctuation and letter case.
2. Elimination of stop words for filtering low discriminating values for retrieval purposes.
3. Stemming of the remaining words to remove prefixes and suffixes.

4. Selection of index terms used as an indexing element. Noun words frequently carry more semantics than adjectives, adverbs and verbs.

5. Construction of term categorization structures.

3.2.2 Document Selection Techniques

The Fisher Criterion, Golub Signal-to-Noise, traditional t-test and Mann-Whitney rank sum statistic were applied to calculate the statistical score, S , for the documents. In these techniques, each document was measured for correlation with the category according to some measuring criteria in the formulas. The documents were ranked according to the score, S , and the top ranked documents were selected. The Fisher Criterion, fisher is a measure that indicates how much the class distributions are separated. The coefficient has the following formula:

$$fisher = \frac{(\mu_1 - \mu_2)^2}{(v_1 + v_2)}$$

Where μ_i is the mean and v_i is the variance of the given document in class i . There were two document classes in this experiment, i.e. the relevant document and the non-relevant documents. The statistic gives higher scores to documents whose mean differ greatly between the two classes, relative to their variances.

Golub used a measure of correlation that emphasizes the "Signal-to-Noise" ratio, *signal to noise*, to rank the documents. It is very similar to the Fisher Criterion but use another related coefficient formula as shown below:

$$signal\ to\ noise = \frac{(\mu_1 - \mu_2)^2}{(\sigma_1 + \sigma_2)}$$

Where μ_i is the mean and σ_i is the standard deviation of the document in class i .

Traditional t-test, t- test assumes that the values of the two document variances are equal. The formula is as follows:

$$ttest = \frac{(\mu_1 - \mu_2)}{\sqrt{((v_p/n_1) + (v_p/n_2))}}$$

Where μ_i is the mean of the document in class I and v_p is the pooled variance.

The Mann-Whitney, mann has the following formula:

$$mann = \frac{n_1 * n_2 * (n_1 + 1)}{2 - r_1}$$

Where n_i is the sizes of class I , and r_1 is the sum of the ranks in class i . The score, S , for each document is thus calculated by using the formula in these statistical techniques.

4. Methodology

This Section presents the analysis carried out on the frequent document set generated from the candidate document set in the training tuples of the document clusters. The splitting criterion is either a split point or split attribute. Text collections for experiments are usually split into two parts: one part for training or building the classifier and a second part for testing the effectiveness of the system.

4.1 Document Classification

Support vector machines (SVM) could provide a learning method that is used for both regression and classification, with a fast algorithm that yields good results for many learning tasks. It is a non-probabilistic binary linear classifier that takes a set of input data and predicts, for each given input, which of the two possible classes comprises the input. Support vectors are the training examples that comprise the support vector machine. Support vector machines cannot handle nominal data, necessitating preprocessing that transforms the nominal data to numerical data. The kernel types supported by this technique are dot, radial, polynomial, neural, analysis of variance (ANOVA), Epanechnikov, Gaussian combination, multiquadric. Dot kernel: The dot kernel is defined by the inner product of. Radial kernel: The radial kernel is defined by $\exp(-g \|x-y\|^2)$ where g is the gamma; it is specified by the kernel gamma parameter. The adjustable parameter gamma plays a major role in the performance of the kernel, and should be carefully tuned to the problem at hand. Polynomial kernel: The polynomial kernel is defined as where x and y are vectors in the input space, i.e. vectors of features computed from training or test samples, and $c \geq 0$ is a free parameter trading off the influence of higher-order versus lower-order terms in the polynomial.

Support Vector Machines are comparatively new category of classification algorithms. An SVM expects a training data set with positive and negative as an input. It then creates a decision limit (the maximal-margin separating limit) between two category and selects the most relevant documents involved in the decision making process (the so called support vectors). The construction of the linear limit is always possible as long as the document is linearly separable. SVMs can use kernels, which provide a nonlinear mapping to a higher dimensional feature space. The dot product has the following formula:

$$K(x, y) = (x \cdot y + 1)^d$$

Where x and y are the vectors of the text, the parameter d is an integer which decides the partition. In the case where d is equals to 1, a linear classification algorithm is generated and it is called the SVM dot product. In the case where d is more than 1, a nonlinear

classification algorithm is generated and it is called SVM quadratic dot product. In this paper, where d is equals to 3, it is called the SVM cubic dot product. The radial basis kernel is as follows,

$$K(x, y) = \exp \left[-\frac{\|x - y\|^2}{2 \sigma^2} \right]$$

Where σ is the median of the Euclidean distances between the members and non members of the document category?

The main advantages of SVMs are that they are robust to outliers, join quickly, and find the optimal decision limit if the document is predictable. Another advantage is that the input document can be mapped into arbitrary high dimensional document clusters where the linear decision limit can be predicted. This mapping allows for higher order of interactions between the samples and can also find correlations between documents. SVMs are also very flexible as they allow for a big variety of kernel functions.

4.2 K-nearest neighbor

The k-nn classification algorithm is a simple algorithm based on a distance metrics between the testing documents and the training documents. The main design of the DCP system is, given a testing document s , and a set of training tuples T containing pairs in the form of (t_i, c_i) where t_i are the values of document i and c_i is the class label of document i . Find k training sample with the most similar documents between t and s , according to distance measure. The class label with the highest rank among the k training sample is assigned to s . The main advantage of k-nn is it has the ability to model very complex target functions by a collection of less complex approximations. It is easy to program and understand. It is robust to noisy training samples.

5. Discussion

In this section, the impact and importance of document selection to the classification performance was first studied. This was carried out by comparing the classification performance by using all documents and document selected by statistical techniques in Section 3.2.2. Two sets of experiments were run: the first were concerned with the effect of the training data samples for feature extraction and the other is testing the features of the test document with the stored features of the trained category. The tools used for training and testing are porter stemmer and Stanford parser. Java is used as a programming language for user interaction with the system. For example suppose we perform retrieval for the query term 'network'. First all documents in training set $[Q]$ tagged with 'network' are selected. Then using the selected documents, word/weight pairs are generated to form a query. This query is used to retrieve the features from the test set $[T]$ to determine the

category. The resulting ranked document list is examined to see where in the ranking documents tagged with 'network' appear. The following procedure gives the steps involved in training and testing phase.

Training

1. Collect some group of documents for each particular category.
 2. Parse the documents by Stanford parser.
 3. Collect a feature set of fourteen features for each document
 4. For each category find the mean values for each feature
- Extracted of all the documents of that particular category to be prepared.

Testing

1. Get an input text file.
2. Parse the document by Stanford parser.
3. Collect a feature set of fourteen features for the document being tested.
4. Match this feature set with each category profile to predict the category of the document.

The system that has been trained and tested was evaluated by human experts. Twenty documents were taken and were given to the classifier system. The same set of documents may also given to human experts. The above discussion shows that the automatic categorizer system development proves to be efficient. The error in category is due to the error made in stemming process. Some terms of category may not stemmed properly and so error occurs in different categories.

References

- [1] Dahl. Improving text retrieval for the routing problem latent semantic indexing. In 17th ACM International Conference on Research and Development in Information Retrieval (SIGIR-94), pages 282–289, 1994.
- [2] D Levis Naïve (bayes) at forty: The independence assumption retrieval. In 10th European Conference on Machine Learning (ECML – 98), pages 4-15, 1998.
- [3] H.Li and K.Yamanishi. Text Classification using esc-based stochastic decision lists. In 8th ACM International Conference on Information and Knowledge Management (CIKM-99), pages 122-130, Kanasa City, USA, 1999.
- [4] Maria-Luiza Antonie, Osmar R. Zaiane, Text Document Categorization by Term Association , in Proc. of the IEEE International Conference on Data Mining (ICDM'2002), pp 19-26 Maebashi City, Japan, December 9 - 12, 2002
- [5] R. Agrawal, T. Imielinski, and A. Swami. M.Ruiz P.Srinivasan Neural networks for text categorization rules between sets items in large databases. In 22nd ACM SIGIR International Conference on Management of Data. Pages 281–282 May1993
- [6] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In 10th European Conference on Machine Learning (ECML-98), pages 137–142, 1998.
- [7] W. Cohen and H. Hirsch. Joins that generalize: text class-search and Development in Information Retrieval (SIGIR-01 4th International Conference on Knowledge Discovery and Data Mining (SigKDD'98), [2] pages 169–173, New York City, USA, 1998.
- [8] R. R. Korfhage. Information Storage and Retrieval. Willey Computer Publishing, 1997.
- [9] G. Salton and M. J. McGill. Introduction to Modern Information Retrieval. Mcw–Gill, 1983.
- [10] R. B. Yates and B. R. Neto. Modern Information Retrieval. Pearson Education, 1999..

