Associative Text Classification For Online Database

V. Srividhya [1] *R.Anitha* [2]

Abstract

Text categorization is regaining interest with the prevalence of digital documents and the wide use of e-mail and web documents, and is becoming a central problem in digital text collections. There have been many approaches to solve this problem, mainly from the machine learning community. This paper explores the use of association rule mining in building a text categorizer. This approach has the advantage of a very fast training phase, less memory usage and the rules of the classifier generated are easy to understand. The investigation leads to conclude that association rule mining is a good and promising strategy for efficient automatic text categorization.

Keywords : Text Categorization, Text Mining, Association Rules and Classification.

1. Introduction

Amazing development of Internet and digital library has triggered a lot of research areas. Text categorization is one of them. Text categorization is a process that group text documents into one or more predefined categories based on their contents [1]. It has wide applications, such as email filtering, category classification for search engines and digital libraries.

[1]Lecturer,Avinashilingam University for Women, Coimbatore-43. Email:vidhyavasu@gmail.com

[2]Director,Department of M.C.A,K.S.R.College of Technology, Tiruchengode-637 215.

Basically there are two stages involved in text categorization. Training stage and testing stage. In training stage, documents are preprocessed and are trained by a learning algorithm to generate classifier. In testing stage, a validation of classifier is performed. There are many traditional learning algorithms to train data. Examples include Decision trees, Naïve-Bayes (NB), Support Vector Machines, k-Nearest Neighbor (kNN), Neural Network (NN), etc. Nowadays, text categorization becomes fundamental given the large number of on-line documents that have to be sorted and grouped. For example large companies could use text classifiers for in-coming e-mail triage and memo categorization. Text classifiers can be used to classify web pages, in-coming emails, memos, news and any other text collection. Building a text classifier usually necessitates a training set consisting of a collection of text documents already associated with topical categories. Once a classifier is built with the training set, a test set, consisting of documents with known categories, is classified and the found class labels compared to the existing categories to determine the effectiveness of classifier.

This paper exploits the use of association rules mining in building categorization system from relatively large training set. The remainder of the paper is organized as follows: Section 2 gives an overview of related work in text categorization and association rule mining. Section 3 shows the new categorization approach. Experimental results are described in section 4. Summarization of research and discussion and research in future direction are narrated in Section 5.

2. Related Work

Many text classifiers have been proposed in the literature using machine learning techniques, probabilistic models, etc. Although a lot of approaches have been proposed, automated text categorization is still major area of research. The use of association rule mining for building classification models is very new. This classification system discovers the strongest association rules in the database and uses them to build categorizer.

In the following subsections a more detailed overview of the related work is presented from both domains: text categorization and association rule mining

2.1 Text Categorization

In the past decade, great attention was paid to the text categorization problem. Most of the text classifiers that were developed and proposed are either machine learning based or statistical based. Classifiers based on probabilistic models have been proposed starting with the first presented in literature by Maron in 1961 and continuing with naïve Bayes [7] that proved to perform well. ID3 and C4.5 are well-known packages whose cores are making use of decision trees to build automatic classifiers [5, 6]. K-nearest neighbor (k-NN) is another technique used in text categorization [11]. Another method to construct a text categorization system is by an inductive learning method. This type of classifier is represented by as set of rules in disjunctive normal form that best cover the training set [4,8,9]. As reported in [12] the use of bigrams improved text categorization accuracy as opposed to unigrams use. In addition, in the last decade neural networks and support Vector Machines (SVM) were used in text categorization and they proved to be powerful tools [10,14].

2.2 Association Rule Mining

2.2.1 Association Rules Generation

Association rule mining has been extensively investigated in the data mining literature. Many efficient algorithms have been proposed. The most popular being is apriori [2] and FP-Tree growth [3]. Association rule mining typically aims at discovering associations between database items in a transactional. Given a set of transactions $D = \{T_1,.., T_n\}$ and a set of items $I= \{i_1,.., i_m\}$ such that any transaction T in D is a set of items in I, an association rule is an implication $A \rightarrow B$ where the antecedent A and the consequent B are subsets of a transaction T in D, and A and B have no common items. For the association rule to be acceptable, the conditional probability of B given A has to be higher than a threshold called minimum confidence. Association rules mining is normally a two-step process, wherein the first step frequent item sets are discovered (i.e., item-sets whose support is no less than a minimum support) and in the second step association rules are derived from the frequent item-sets.

2.2.2 Associative Classifiers

Besides the classification methods, associative text categorization, and a new method that builds associative general classifiers. In this case the association rule mining represents the learning method.

The main idea behind this approach is to discover strong patterns that are associated with class labels. The next step is to take advantage of these patterns such that a classifier is built and new objects are categorized in the proper classes.

3. Building An Associative Text Classifier

In this paper, a method to build a categorization system that merges association rule mining task with the classification problem is presented. Given a data collection, a number of steps are followed until the

classification model is found. Data preprocessing represents the first step. The next step in building the associative classifier is the generation of association rules using FP-growth algorithm. The last stage in this process is represented by the use of the association rules set in the prediction of classes for new documents.

3.1 Association Rule Generation : The FP-Growth Algorithm

The FP-growth algorithm is currently one of the fastest approaches to frequent item set mining. The association rules discovered in this stage of the process is further processed to build the associative classifier.

The main bottleneck of the Aprioi-like methods is at the candidate set generation and test. This problem was dealt with by introducing a novel, compact data structure, called Frequent Pattern tree, or FP-tree then based on this structure, an FP-tree-based pattern fragment growth method was developed, FP-growth. Figure 1 presents the pseudo code of FP-Growth algorithm.

The basic principle of FP-Growth is to work in a divide and conquer manner. Compared to Apriori, the FP-Growth algorithm requires only two scans on the database. It first computes a list of frequent items sorted by frequency in descending order (F-List) during its first database scan. In its second scan, the database is compressed into a FP-tree. Then FP-Growth starts to mine the FP-tree for each item whose support is larger than ξ by recursively building its conditional FP-tree. The algorithm performs mining recursively on FP-tree. The problem of finding frequent itemsets is converted to searching and constructing trees recursively.

3.2 Classificaiton Process

Given a collection of documents, the first step is to index them to produce document representations. In the full text logical view, a representation of a document d_j is the set of all its terms (or words). Each term of the document representation is considered as a separate variable or feature. Bag-of-wors technique is used for this purpose. From this, a subset of the terms to represent the documents is selected, through a process called feature selection, to reduce the document representations. For this purpose, preprocessing techniques like stop-word elimination, stemming and TF/IDF are used. Then the FP-Growth Algorithm is used to generate a set of association rules, which are used during the classification process.

```
Input : database DB, minimum support
Output : the complete set of frequent patterns
Method : FPGrowth(DB, ξ)
Define and clear F-List : F[ ];
for each Document T_i in DB do
    for each Term a_j in T_i do
        F[a_j] ++;
    end
end
Sort F[];
Define and clear the root of FP-tree : r_i
for each Document Ti in DB do
    Make T_i ordered according to F;
    Call ConstructT ree(T_i; r);
end
for each term a_i in I do
    Call Growth(r, a_i, ξ);
end
```

Figure 1: FP-Growth Algorithm

To improve the effectiveness of the classification process, the classifier goes through a process of fine tunning its internal parameters. This is accomplished using a training set. Once its parameters have been fine tunned, the classifier is used to classify the new documents. This process is discussed in the next section. The final phase

of the process is to evaluate the effectiveness of the classification. The evaluation of a classifier is done by comparing the results with standard machine learning classifiers This process is pictorically given in Fig. 2.

3.3 Prediction of Classes Associated with New Documents

The set of rules that were selected represent the actual classifier. This categorizer will be used to predict to which class a new document will be attached. Given a new document, the classification process searches in this set of rules for finding those classes that are the closest to be attached with the document presented for categorization. This subsection discusses the approach for labeling new documents based on the set of association rules that forms the classifier.

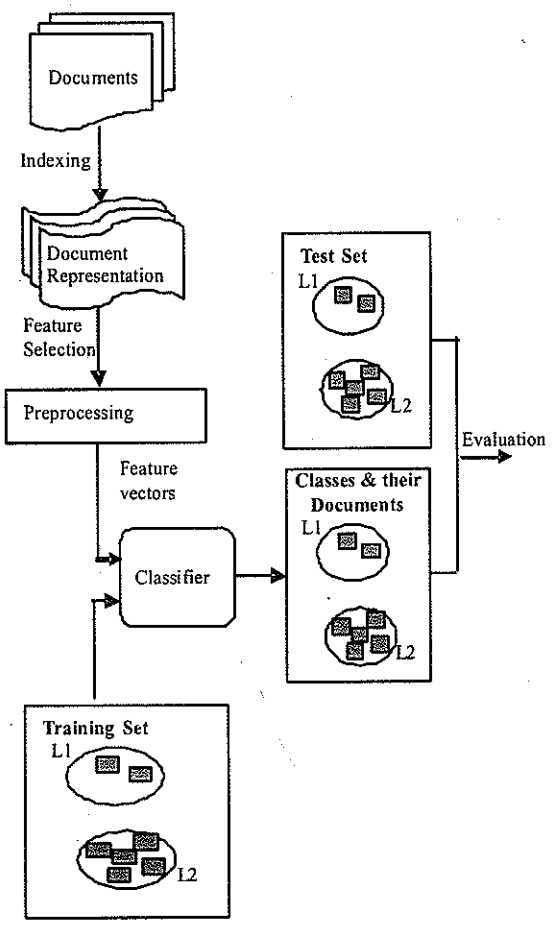

Figure 2 : The Text Classification Process Used In This Paper

Given a document to classify, the terms in the document would yield a list of applicable rules. If the applicable rules are grouped by category in their consequent part and the groups are ordered by the sum of rules confidences, the ordered groups would indicate the most significant categories that should be attached to the document to be classified. This order category is named as dominance factor δ. The dominance factor allows us to select among the candidate categories only the most significant. When δ. is set to a certain percentage a threshold is computed as the sum of rules confidences for the most dominate category times the value of the dominance factor. Then, only those categories that exceed this threshold are selected. The function (TakeKClasses) selects the most k significant classes in the classification algorithm. The algorithm used is given in Fig. 3.

Algorithm : Classification of a new object

Input : A new object to be classified o; the Associative classifier (ARC); the dominance factor δ; the confidence threshold T

Output : Categories attached to the new object

(1) s <- ∅ /*set of rules that match */

(2) **foreach** rule r in ARC (the sorted set of rules)

(3) **if** (r ⊂ o) {count++}

(4) **if** (count == 1)

(5) fi.conf <- r.conf /*keep the first rule confidence*/

(6) S <- S Ù r

(7) **else if** (r.conf > fi.conf-T)

(8) S <- S Ù r

(9) **else** exit

(10) divide S in subsets by category : S1,S2…Sn

(11) **foreach** subset S1,S2…Sn

(12) sum the confidences of rules and divide by the number of rules in S_k

(13) **if** it is single class classification

(14) put the new document in the class that has the highest confidence sum

(15) **else** /*multi-class classification*/

(16) TakeKClasses(S, δ)

(17) assign these k classes to the new document

Figure 3 : Classification of a New Object

4. Experimental Result

4.1 Experiment Data

The Reuters 21578 text collection was used as benchmarks in evaluating the system. Reuters 21578 is split into two parts: one part for training and a second part for testing. The ModApte version of split is used in this research. This split leads corpus of 12,202 documents consisting of 9,603 training documents and 3,299 testing documents. There are 135 topics to which documents are assigned [15]. Finally The classifier was tested with ten most populated categories with largest number of documents assigned to them in training set. As most of the researches [13] have following this strategy, the present wok uses the same approach for evaluation so that the results can be compared with the standard techniques. By retraining only the ten most populated categories, there are total of 6488 training documents and 2425 testing documents.

4.2 Experimental Results

When dealing with multiple classes there are two possible ways of averaging these measures, namely macro average and micro average. In micro average for all classes, an average of all classes is computed and the performance measure obtained there in. The macro-average weights equally all the classes, regardless of how many documents belonging to it. The micro average weights equally all the documents, thus favoring the performance of all classes.

Table 1 shows a comparison between the proposed ARC-FG(Association Rule based categorizer - Frequent Growth) classifier and other well-known methods. The measures used are precision/recall-breakeven point; micro average and macro average on ten most populate Reuters categories. The proposed system proves to perform well as compared to the other methods. In general, the functioning of the proposed algorithm is in par with the existing state of the art text classifiers. In addition to these results, the system has two more features. First it is very fast in both training and testing documents. Second, it produces readable and understandable rules that can be easily modified by humans.

5. Conclusion And Future Work

This paper approaches the probem of online text categorization using association rules. In particular, the study involves the application of FP-Growth algorithm to online news classification. The study provides evidence that association rule mining can be used for the construction of fast and effective classifiers for automatic text categorization. One major advantage of the association rule based classifier is that it does not assume that terms are independent and its training is relatively fast.

Table 1 : Precision/Recall Breakeven Point On Ten Most Populated Reuters Categories For ARC-FG and Most Known Classifiers

category	ARC-FG			Bayes	Rocchio	C4.5	k-NN	SVM
	10%	15%	20%					
acq	86.4	86.8	87.1	91.5	92.1	85.3	92.0	94.5
corn	84.6	84.2	83.9	47.3	62.2	87.7	77.9	85.4
crude	82.1	81.7	81.4	81.0	81.5	75.5	85.7	87.7
earn	77.3	77.7	78.0	95.9	96.1	96.1	97.3	98.3
grain	70.6	70.2	69.9	72.5	79.5	89.1	82.2	91.6
interest	81.9	81.5	81.2	58.0	72.5	49.1	74.0	70.0
money-fx	80.1	80.5	80.8	62.9	67.6	69.4	78.2	73.1
ship	74.2	73.8	73.5	78.7	83.1	80.9	79.2	85.1
trade	95.5	95.1	94.8	50.0	77.4	59.2	77.4	75.1
wheat	74.9	75.3	75.6	60.6	79.4	85.5	76.6	84.5
micro-avg	86.6	86.6	86.5	72.0	79.9	79.4	82.3	85.4
macro-avg	80.8	80.7	80.6	65.21	79.14	`77.78	82.05	84.58

Furthermore, the rules are human understandable and easy to be maintained. Feature selection can be done by adding the weight of each term in the documents and pruning the terms with lower weight. The feature selection will reduce the number of terms as well as reduce the noisy of the terms. The feature selection techniques such as latent semantic analysis could improve the results.

References

[1] K.Aas and A.Eikvil, *"Text categorization: A survey"*, Technical report, Norwegian Computing Center, June, 1999.

[2] Agrawal, R.Srikant, *"Fast Algorithm for Mining Association Rules"*, Proc. VLDB Conf. Santiago, Chile, 1994.

[3] Han. J, Pei.J, Yin.Y, *"Mining Frequent Patterns without Candidate Generation"*, Proc. ACM-SIGMOD, Dallas, 2000.

[4] C.Apte, F.Damerau and S.Weiss, *"Automated learning of decision rules for text categorization"*, ACM Transactions on Information systems,1994.

[5] W.Cohen and H.Hirsch, *"Joins that generalize: text classification using whirl"*, In 4th International Conference on Knowledge Discovery and Data Mining (SigKDD'98), New York City, USA.

[6] W.Cohen and Y.Singer, *"Context-sensitive learning methods for text categorization"*, ACM Transactions on Information Systems, 1999.

[7] D. Lewis, *"Naive (bayes) at forty: The independence assumption in information retrieval"*, In 10th European Conference on Machine Learning.

[8] H.Li and K.Yamanishi, *"Text classification using esc-based stochastic decision list"*, In 8th ACM International Conference on Information and knowledge Management.

[9] B.Liu, W.Hsu and Y.Ma, *"Integrating classification and association rule mining"*, In ACM Int.Conf.on knowledge Discovery and Data Mining.

[10] M.Ruiz and P.Srinivasa, *"Neural networks for text categorization"*, In 22nd ACM SIGIR International Conference on Information Retrieval, Berkeley.CA, USA.

[11] Y.Yang, *"An evaluation of statistical approaches to text categorization"*, Technical Report CMU-CS-97-127, Carnegie mellon University, April 1997.

[12] C.M.Tan, Y.F.Wang and C.D.Lee, *"The use of bigrams to enhance text categorization"*, Journal of Information processing and management, 2002.

[13] F.Sebastiani, *"Machine learning in automated text categorization"*, Technical Report IEI-B4-31-1999, Consiglio Nazionale delle Ricerche, Pisa, Italy, 1999.

[14] Y.Yang and X.Liu, *"A re-examination of text categorization methods"*, In 22nd ACM International Conferernce on Research and Development in Information Retrieval (SIGIR-99), PP. 42-49,Berkeley, US, 1999.

[15] O.R.Zaiane and M.L. Antonie, *"Classifying text documents by associating terms with text categories"*, In Thirteenth Australasian Database Conference (ADC' 02), PP. 215-222, Melbourne, Australia, January 2002.

Author's Biography

Mrs.V. Srividhya received her M.Sc (Computer Science) from Madurai Kamaraj University in 1996. She obtained her M.Phil(Compute Science) from Bharathiar University in 2000. She is currently working towards the Ph.D degree with the specialization of Data mining. Since 2007, she has been associated with the Avinashilingam University for Women, Coimbatore, where she is currently an Asst.Professor in the Department of Computer Science. Her current research interests are in the area of Text mining and Web mining. She has presented and published number of papers in different International and National Conferences and journals.

Dr. R.Anitha obtained her Ph.D. from Periyar University, Salem. She is currently working as a Professor & Director, Department of MCA, K.S. Rangasamy college of Technology, Tiruchengode. Her area of Specialization is Digital Image Processing. She has presented and published number of papers in different International and National Conferences and journals.