Automatic Text Classification using Modified Centroid Classifier

Mahmoud Elmarhumy
Faculty of Engineering,
University of Tokushima
2-1 Minamijosanjima
Tokushima, Japan 770-8506
m_elmarhoumy@yahoo.com

Mohamed Abdel Fattah
FIE, Helwan University,
Cairo, Egypt
mohafi@is.tokushima-u.ac.jp

Fuji Ren
Beijing University of Posts
& Telecommunications
Beijing, 100088, China
ren@is.tokushima-u.ac.jp

Abstract:

This work proposes an approach to address the problem of inductive bias or model misfit incurred by the centroid classifier assumption to enhance the automatic text classification task. This approach is a trainable classifier, which takes into account *tfidf* as a text feature. The main idea of the proposed approach is to take advantage of the most similar training errors to the classification model to successively update it based on a certain threshold. The proposed approach is simple to implement and flexible. The proposed approach performance is measured at several threshold values on the Reuters-21578 text categorization test collection. The experimental results show that the proposed approach can improve the performance of centroid classifier.

Keywords:

Text classification; text categorization; centroid classifier; Data mining.

1. Introduction

Text classification/categorization is defined as the task of classifying documents into a fixed number of predefined categories. In most cases, usual text string representation of a document is transformed into a numeric feature vector at a pre-processing step and then the vector is provided to a learning algorithm as input. Each component of the vector represents the value of one attribute of the document.

The rapid growth of the Internet has been increased the number of online documents available. This has been led to the development of automated text and document classification systems that are capable of automatically organizing and classifying documents. In particular, automatic text categorization has been extensively studied recently. This categorization problem is usually viewed as supervised learning, where the goal is to assign predefined category labels to unlabeled documents based on the likelihood inferred from the training set of labeled documents. Many approaches have been applied, including entroid Classifier, Bayesian probabilistic approaches [11] nearest neighbor, neural networks, decision trees [1][2], inductive rule learning [5], support vector machines [9], maximum entropy, boosting, multivariate regression models [8], symbolic rule learning [3][6], Rocchio classifiers [12] and linear discriminate projection [4]. In general, much research has been done on comparing the performance of different text categorization techniques using several datasets. Among these models, a variant of linear models called a centroid-based method is attractive since it has relatively less computation than other methods in both the learning and classification stages. The traditional centroid-based method [10] can be viewed as a specialization of so-called Rocchio method [12] and used in several works on text categorization [11]. Based on the vector space model, a centroid-based method computes beforehand, for each category, an explicit profile (or class prototype), which is a centroid vector for all positive training documents of that category. The classification task is to find the most similar class to the vector of the document we would like to classify, for example by the means of cosine similarity. This type of classifiers is easy to implement and effective in computation. However, it often suffers from the inductive bias or model misfit [7] incurred by its assumption. In substance, Centroid Classifier makes a simple hypothesis that a given document should be assigned a particular class if the similarity of this document to the centroid of the class is the largest. However, this supposition is often violated (misfit) when exists a document from a certain class sharing more similarity with the centroid of another class than that of its class. The more serious the model misfit, the poorer the classification performance will be (Tan, 2008). To avoid the problem of inductive bias or model misfit, we propose an approach that takes advantage of the most similar training errors to the classification model to successively update it based on a certain threshold. The proposed approach moves the centroid of each class by a certain distance based on the miss-classified documents of each class that have maximum similarities with their class. Considering only training errors that have maximum similarities with their class increases the classification accuracy. However, taking all training errors into account will move each class centroid by a

978-1-4244-4538-7/09/$25.00 ©2009 IEEE

large distance that makes all centroids incorrectly distributed.

The rest of the paper is organized as follows: section 2 presents the centroid classifier basics, section 3 is about the proposed automatic classification model, section 4 shows the experimental results and finally section 5 presents conclusions and future work.

2. Centroid Classifier Basics

Given a set of classes $C = \{c_1, c_2, \dots c_m\}$ and a set of training documents $D = \{d_1, d_2, \dots d_N\}$ where each training document d_i is assigned to one or more classes, text categorization is a task to use this given information to find one or more suitable categories for a new document. In a vector space model, a document (or a class) is represented by a vector based on the weight of each term in the document or class.

Most research works applied term frequency (*tf*) and inverse document frequency (*idf*) in the form of *tfidf* to weight a term in a document. The term frequency in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents (which may have a higher term count regardless of the actual importance of that term in the document) to give a measure of the importance of the term *ti* within the particular document *dj*. Thus we have the term frequency, defined as follows.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \qquad (1)$$

Where $n_{i,j}$ is the number of occurrences of the considered term in document d_j, and the denominator is the sum of number of occurrences of all terms in document d_j. The inverse document frequency is a measure of the general importance of the term (obtained by dividing the number of all documents by the number of documents containing the term, and then taking the logarithm of that quotient).

$$idf_i = \log \frac{|D|}{|\{d : t_i \in d\}|} \qquad (2)$$

With $|D|$ = total number of documents in the corpus and $|\{d : t_i \in d\}|$ = number of documents where the term t_i appears (that is $\mathrm{n}_{\mathrm{i,j}} \neq 0$). Then

$$tfidf_{i,j} = tf_{i,j} \times idf_i \qquad (3)$$

A high weight in *tfidf* is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms.

The documents are represented using vector space model (*VSM*). In this model, each document *d* is considered to be a vector in the term-space and each term of this vector is *tfidf*.

To calculate the centroid associated with a certain class c_i, we sum document vectors in this class as follows:

$$C_{i\_sum} = \sum_{d \in C_i} d \qquad (4)$$

To achieve the centroid c_i, we normalize c_{i_sum} as follows:

$$C_i = \frac{C_{i\_sum}}{\| C_{i\_sum} \|_2} \qquad (5)$$

Where $\| C_{i\_sum} \|_2$ denotes the 2-norm of vector $\| C_{i\_sum} \|$.

Based on the derived class vectors, a query document is classified by calculating the similarity between the query vector representing the query document and each class vector, and then selecting the most similar class as the resultant class for that query document. To this end, the inner-product is used for similarity measurement. It is calculated by the dot product between the class vector and the query vector. Therefore, the test document d_t will be assigned to the class c_i whose class prototype vector is the most similar to the query vector of the test document as follows:

$$c_i = \arg\max_{c_k \in C} sim(d_t, c_k) = \arg\max_{c_k \in C} d_t . c_k$$

3. The Proposed Automatic Classifier Model

All In centroid classifier, it is assumed that a given document should be assigned a particular class if the similarity between this document and the centroid of its true class is the largest. Nevertheless, this supposition is often violated when we find documents from a certain class sharing more similarities with the centroids of other classes.

Let us consider a two class text data as shown in Fig. 1. Class A spreads as triangle shape; while class B spreads as circle shape. Obviously, the examples d1a to d3a are correctly classified as category A since they share more similarity with centroid A rather than centroid B. On the other hand, d4a to d10a will be misclassified into class B since they share more similarity with centroid B rather than centroid A. Adding the training errors (d5a to d10a) to centroid A to adjust its prototype vector will move centroid A towards centroid B by a large distance. This will make centroids A and B closed to each other. This definitely will deteriorate the total system performance. Moreover, adding the training errors (d4a to d10a) to centroid A and subtracting them from centroid B will deteriorate the total system performance as well. Since as shown in the examples of fig. 1, the centroid A will move toward the original documents of class B. While centroid B will move away from its original documents that makes some of class B original documents misclassified into class A.

The same problem occurs for class B documents that share more similarity with centroid A rather than centroid B. To overcome the drawback of the traditional centroid, we propose the modified centroid classifier model. In the proposed model, we add the most similar training errors belonging to a certain class to its centroid to update it and discard the training errors that have low similarities with their class based on a certain threshold value according to the following formula:

$$C_{i\_\mathrm{mod}ified} = C_i + \frac{\sum_{d \in class\, i\, \&\, classified\ as\ other\ categories\ and\ has\ similarity\ with\ C_i > threshold} d}{\| \sum_{d \in class\, i\, \&\, classified\ as\ other\ categories\ and\ has\ similarity\ with\ C_i > threshold} d \|_2} \tag{7}$$

Since C_i is calculated from equation 5. Based on formula 7, we first need to pick out the total misclassified examples by categorizing all training documents and then update the corresponding centroid by adding the most similar misclassified training errors to their correct classes based on a certain threshold value.

4. Experimental Results

4.1. The training and testing data

The Reuters-21578 collection has been exploited as training and testing data. It is distributed in 22 files. Each of the first 21 files (reut2-000.sgm through reut2-020.sgm) contains 1000 documents, while the last (reut2-021.sgm) contains 578 documents. This corpus consists of 21,578 newswire stories about financial categories collected from Reuters during 1987. For each document, a human indexer decided the categories from which sets that document belonged to. There are 135 different categories, which are overlapping and non-exhaustive, and there are relationships among the categories. We have used this corpus' subset that consisting of the ten most frequent categories. Half of the Reuters-21578 data corpus was used as training data and the other half was used as testing data.

Here we use classification accuracy for evaluation. Different measures, such as precision-recall graphs and F measure have been used in the literature. However, since our goal in text categorization is to achieve low misclassification rates and high separation between different classes on a test set, we thought that accuracy is the best measure of performance.

4.2. The traditional centroid classifier

We have exploited the traditional centroid classifier as described in section 2. Based on formula 5 then formula 6, the testing documents are classified. Table 1 shows the document classification accuracy associated with the traditional centroid classifier.

4.3 The proposed approach

To overcome the drawback of the previously mentioned approach, we exploit the modified centroid classifier model. In this model, we add the most similar training errors belonging to a certain class to its centroid to update it and discard the training errors that have low similarities with their class based on a certain threshold value according to formula 7. Using formula 7 then formula 6 after substituting C_i with $C_{i\_\mathrm{mod}ified}$, may enhance the total system performance since the class centroids will move small distances to include some misclassified documents in addition to the original correctly classified documents. Table 2 shows the document classification accuracy when we use formula 7 for threshold value = 0.20.

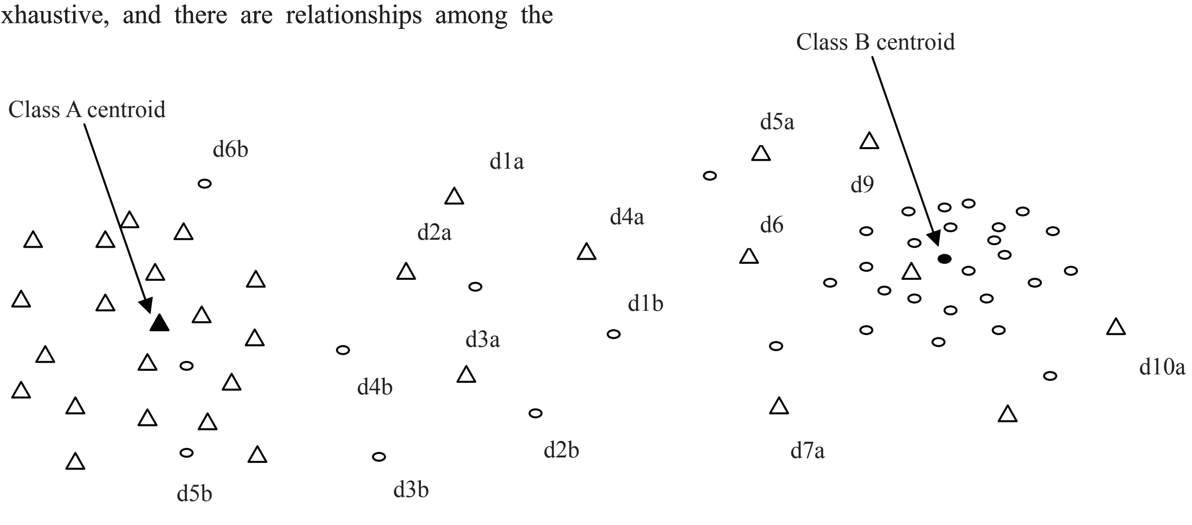

Fig. 1. **A two class text data**

Table 1 The document classification accuracy using traditional centroid classifier

Category	Acquisition	Corn	Crude	Earn	Grain
Accuracy	98.5%	80.6%	81.1%	91.7%	46.6%
Category	Interest	Money-fx	Ship	Trade	Wheat
Accuracy	88.3%	70.8%	78.4%	93.4%	89.0%
Total	Acquisition+Corn+Crude+Earn+Grain+Interest+Money-fx+Ship+Trade+Wheat				
Accuracy	91.5%				

Table 2 The document classification accuracy using proposed approach with threshold value = 0.20

Category	Acquisition	Corn	Crude	Earn	Grain
Accuracy	95.5%	80.6%	84.0%	96.4%	46.7%
Category	Interest	Money-fx	Ship	Trade	Wheat
Accuracy	87.9%	70.3%	77.6%	93.4%	89.0%
Total	Acquisition+Corn+Crude+Earn+Grain+Interest+Money-fx+Ship+Trade+Wheat				
Accuracy	92.5%				

5. Conclusion and Future Work

In this paper, we have investigated different approaches for automatic text classification. Firstly, we have exploited the traditional centroid classifier as a baseline model. Then, we proposed the modified centroid classifier to improve the system performance based on a certain threshold value.

In the future work, we will exploit this approach with some other approaches to construct a hybrid model to improve automatic text classification.

References

[1] Aas K. & Eikvil L., “Text categorization”: a survey. Norwegian Computing Center. *Available from* *http://citeseer.ist.psu.edu/aas99text.html*, 1999.

[2] Apte C., Damerau F., & Weiss, S., “Text mining with decision rules and decision trees”, Proceedings of the workshop with conference on automated learning and discovery: Learning from text and the web, 1998.

[3] Apte, C., Damerau, F., & Weiss, S. M., “Automated learning of decision rules for text categorization”, ACM Transactions on Information Systems, Vol. 12, No.3,pp. 233–251, 1994.

[4] Chakrabarti, S., Roy, S., & Soundalgekar, M. V., “Fast and accurate text classification via multiple linear discriminant projections”, Proceedings of the 28th international conference on very large data bases (VLDB’02). San Francisco, CA: Morgan Kaufmann, 2002.

[5] Cohen, W. W., & Singer, Y., “Context-sensitive learning methods for text categorization”, ACM Transactions on Information Systems, Vol. 17, No. 2, pp. 141–173, 1999.

[6] Cohen, W. W., & Singer, Y, “Context-sensitive learning methods for text categorization”, *SIGIR-96,* 1996.

[7] Dumais, S., Platt, J., Heckerman, D., & Sahami, M., “Inductive learning algorithms and representations for text categorization”, *CIKM,* 1998.

[8] Fuhr, N., Hartmanna, S., Lustig, G., Schwantner, M., & Tzeras, K., “A rule-based multi-stage indexing system for large subject fields”, Proceedings of the RIAO’91, pp. 606–623, 1991.

[9] Godbole, S., Sarawagi, S., & Chakrabarti, S., “Scaling multi-class support vector machine using inter-class confusion”, Proceedings of the 8th ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD 2002, pp. 513–518, 2002.

[10] Han, E.-H., Karypis, G., “Centroid-based document classification: analysis and experimental results”, Principles of Data Mining and Knowledge Discovery, 2000, pp. 424–431, 2000.

[11] Ittner, D.J., Lewis, D.D., Ahn, D.D., “Text categorization of low quality images”, Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, USA, 1995, pp. 301–315, 1995.

[12] Rocchio J.J., Jr., “Relevance feedback in information retrieval”, G. Salton (Ed.), The SMART REtrieval System: Experiments in Automatic Document Processing, Prentice-Hall, *Englewood Cliffs, NJ, 1971, pp. 313–323,* 1971.