An Effective Class-centroid-based Dimension Reduction Method for Text Classification

Guansong Pang
School of Management, Guangdong University of Foreign Studies
Guangzhou 510006, China
pangguansong@163.com

Huidong Jin
CSIRO Mathematics, Informatics and Statistics
Canberra 2601, Australia
Warren.Jin@csiro.au

Shengyi Jiang
School of Informatics, Guangdong University of Foreign Studies
Guangzhou 510006, China
jiangshengyi@163.com

ABSTRACT

Motivated by the effectiveness of centroid-based text classification techniques, we propose a classification-oriented class-centroid-based dimension reduction (*DR*) method, called *CentroidDR*. Basically, *CentroidDR* projects high-dimensional documents into a low-dimensional space spanned by class centroids. On this class-centroid-based space, the centroid-based classifier essentially becomes *CentroidDR* plus a simple linear classifier. Other classification techniques, such as K-Nearest Neighbor (*KNN*) classifiers, can be used to replace the simple linear classifier to form much more effective text classification algorithms. Though *CentroidDR* is simple, non-parametric and runs in linear time, preliminary experimental results show that it can improve the accuracy of the classifiers and perform better than general *DR* methods such as Latent Semantic Indexing (*LSI*).

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining

Keywords

Text Classification, Dimension Reduction, Class Centroid

1. INTRODUCTION

One of the big challenges in web page classification, email spam detection or text classification in general is that very high data dimensionality renders various classification techniques less effective. Dimension reduction (*DR*) is one of the most effective techniques to tackle this challenge. Some general *DR* methods have been developed, e.g., Latent Semantic Indexing (*LSI*) [1]. The main defect of using *general DR techniques* for classification is that they are blind to the latent relation between classes and terms. Therefore, the new space found does not necessarily provide the best separation of documents of the underlying class-distribution. In order to use class labels to adapt unsupervised methods such as *LSI*, Supervised Latent Semantic Indexing (*SLSI*) has been proposed [1]. The experimental results in [1] show the effectiveness of *SLSI*, but the improvements are relatively limited. Furthermore, these general *DR* techniques normally require parameter tuning and expensive computation.

In this paper, for text classification, we propose an effective linear-time class-centroid-based *DR* method (*CentroidDR*). The underlying hypothesis of our method is that documents (except outliers and noisy documents) are normally closer to their inherent class centroid rather than other centroids. This hypothesis has been demonstrated by the effectiveness of centroid-based text classifiers (namely *Centroid*) [2]. However, unlike *Centroid* that find the most similar class centroid for test documents in order to perform classification, our method aims to use class centroids to reduce the dimensionality of documents. Basically, *CentroidDR* projects high-dimensional documents into a low-dimensional space spanned by class centroids. On this class-centroid-based space, intuitively, *Centroid* is essentially *CentroidDR* plus a simple linear classifier. By combining *CentroidDR* with a sophisticated (e.g., non-linear) classifier, we can construct a more effective algorithm to find the better classification boundary on this new space. Therefore, widely used classification techniques such as *KNN* can use *CentroidDR* to strengthen their classification performance, as well as to benefit from the computational efficiency of low-dimensional data.

2. PROPOSED METHOD: *CentroidDR*

Given a set of *N* documents $D_{doc} = \{(d_1, y_1), (d_2, y_2), \cdots, (d_N, y_N)\}$, where $d_i \in \Re^n$ (a vector represented by *Vector Space Model* (*VSM*) and weighted by *term frequency* (*TF*) and *inverse document frequency* (*IDF*) [2, 3]) represents document *i, n* is the number of different terms in D_{doc}, and $y_i \in \{C_1, C_2, \cdots, C_l\}$ a set of predefined classes. A class centroid is calculated as follows:

$$centroid_j = \frac{1}{|C_j|} \sum_{d_i \in C_j} d_i \tag{1}$$

So $centroid_j$ denotes the centroid of the class C_j, $j = 1,2,\cdots,l$. We project documents onto the new space via formula (2).

$$x_i^{(j)} = cosine(d_i, centroid_j) = \frac{\sum_{t=1}^{n} d_i^{(t)} \times centroid_j^{(t)}}{\sqrt{\sum_{t=1}^{n} \left(d_i^{(t)}\right)^2} \times \sqrt{\sum_{t=1}^{n} \left(centroid_j^{(t)}\right)^2}} \tag{2}$$

where $i = 1,2,\cdots,N$, $j = 1,2,\cdots,l$ and $t = 1,2,\cdots,n$. Details of *CentroidDR* are given as follows:

Input: Given a set of N documents D_{doc}.

Output: Projected data $D_{reduced} = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$, where $x_i \in \Re^l$, $y_i \in \{C_1, C_2, \cdots, C_l\}$, $i = 1,2,\cdots,N$.

Procedure:

(1) Compute the centroid of each class via formula (1).

(2) For each document d_i, compute the similarities between the document and all the centroids, and assign the similarity values to the dimension values of its projected data x_i via formula (2).

(3) Obtain $D_{reduced}$. The coordinates of the original documents in the reduced space are the similarities obtained in the step (2), and their class labels are retained from their original documents.

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

We illustrate the working scheme of *CentroidDR* on two classes of documents from Reuters-21578 corpus, *money-fx* and *money-supply*, which contain 676 documents and 12315 different terms in total. We plot the *Centroid* and *KNN* classification results on the projected data in Figure 1, where the *red dashed line* indicates the classification boundary of *Centroid*, which misclassifies 40 documents. The classification could be improved. We use the *green contour* to represent the KNN classification boundary. The figure shows that *KNN* ($K = 5$) works on the 2-*D* data better than *Centroid*, and only misclassifies 15 documents.

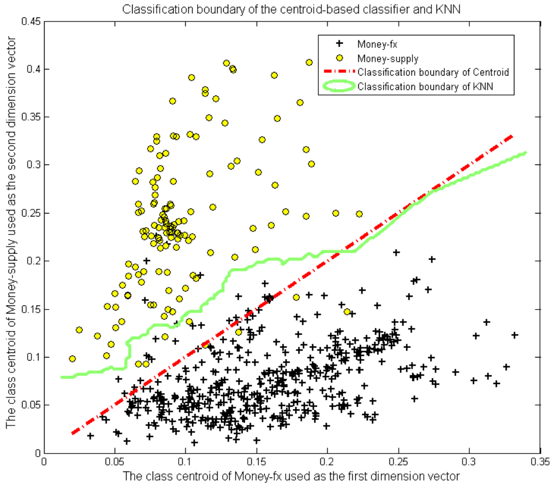

Figure 1. Visualization of classification results

3. EXPERIMENTS

Following [1, 3], we performed experiments on two subsets (i.e., top 7 and 8 categories) of Reuters-21578 (namely *R7* and *R8*) and the whole set of Fudan text classification corpus (namely *Fudan20*). *MicF*$_1$ (short for micro averaging F_1) and *MacF*$_1$ (short for macro averaging F_1) [1-3] were used as metrics.

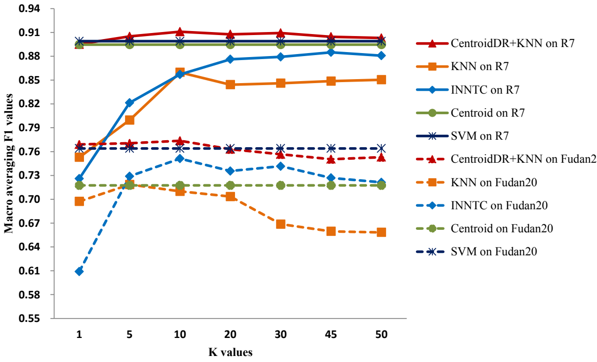

Figure 2. *MacF*$_1$ on *R7* and *Fudan20* with varying *K* values

By using *CentroidDR*, the documents of *R7* and *Fudan20C* are projected from 60345-*D* and 335664-*D* term-based space onto a class-centroid-based space with only 7 and 20 dimensions respectively. To evaluate our method, we applied *KNN* for classification on these two corpora on the class-centroid-based space by comparing to *KNN*, *Centroid*, *INNTC* (an improved *KNN* method [3]) and Support Vector Machines (*SVM* with linear kernel [3]) on the term-based space. We obtain similar trends in *MicF*$_1$ and *MacF*$_1$ values. Here we report the *MacF*$_1$ results in Figure 2. It is clear that "*CentroidDR+KNN*" consistently outperforms *INNTC*, *Centroid* and is comparable to *SVM* using different *K* values on both *R7* and *Fudan20*, though *KNN* has less effective performance than these classifiers on the term space.

We compared *CentroidDR* directly with some best experimental results of its counterparts reported in [1]. *LSI* achieved the best performance when projecting the documents of *R8* into 32 dimensions. We derive the best results for this case, i.e., *MicF*$_1$ and *MacF*$_1$ are about 0.9150 and 0.8000 respectively [1]. *SLSI* obtained the same performance as *LSI* when projecting documents onto 11 dimensions. These results were based on the *KNN* classifier. In our experiment, we combined *CentroidDR* with *KNN* to do dimension reduction and classification on the same dataset, with the documents projected onto an 8-*D* space. The results show that "*CentroidDR +KNN*" consistently outperforms *LSI* and *SLSI* with 0.9188 and 0.8438 in *MicF*$_1$ and *MacF*$_1$ respectively.

CentroidDR consists of two main stages: class centroid generation and class-centroid-based projection. We only need to scan the text collection twice for these two stages. Therefore, *CentroidDR* has the linear time complexity $O(Nn)$, comparing with $O(Nn^2)$ of *LSI*. Also, classifiers can benefit from the computational efficiency of low-dimensional data. For example, *KNN*, averaged over various different *K* values, consumes about 9 and 55 times longer computation time for *online* classification than "*CentroidDR +KNN*" on *R7* and *Fudan20C* respectively.

4. DISCUSSIONS

We proposed the method *CentroidDR*. Compared to other advanced *DR* methods, e.g., *LSI*, *SLSI*, *CentroidDR* is much more feasible and scalable, as it is linear-time, non-parametric and easy-to-implement. Combining *CentroidDR* with learning methods such as *KNN* to conduct text classification can project the high-dimensional documents into a very low-dimensional space, and further improve the accuracy of the classifiers, as well as their computation efficiency. More experiments on other corpora and combining other classifiers with *CentroidDR* are under way. We are also investigating the effectiveness of *CentroidDR* on data sets with a large number of classes or with very skewed classes.

5. ACKNOWLEDGMENTS

We thank the anonymous reviewers. This work was supported in part by the NSF of China (No. 61070061, 61202271), Guangdong Province Colleges and Universities Discipline Construction Project (No.2012KJCX0049), and National Key Technologies R&D Program Project (No.2012BAH02F03).

6. REFERENCES

[1] Sun, J., Chen, Z. et al. 2004. Supervised latent semantic indexing for document categorization. ICDM'04, 535-538.

[2] Han, E., Karypis, G. 2000. Centroid-based document classification: analysis and experimental results. PKDD'00, 116-123.

[3] Jiang, S., Pang, G. et al. 2012. An improved K-nearest-neighbor algorithm for text categorization. Expert Systems with Applications, 39(1), 1503-1509.