FEATURE SELECTION USING SINGULAR VALUE DECOMPOSITION AND ORTHOGONAL CENTROID FEATURE SELECTION FOR TEXT CLASSIFICATION

Hoan Dau Manh[1]

[1]*Quang Binh University, Viet Nam*

Abstract

Text mining is a narrow research field of data mining, which focuses on discovering new information from text document collections, mainly by using techniques from data mining, machine learning, natural language processing and information retrieval. Text classification is the process of analyzing text content and then giving decision whether this text can belong to one group, many groups or it does not belong to the text group which is defined before. On over the world, there have been many effective researches on this problem, especially on texts in English. However, there have been few researches on Vietnamese texts. Moreover, these researching results and applications are still limited partly due to the typical characteristics of Vietnamese language in term of words and sentences and there are many words with many meanings in many different contexts. Text classification problem is the one with many featues, thus to improve the effectiveness of text classification is the aim of may researchers. In this research, the author constructs two methods of feature selection: singular value decomposition and optimal orthogonal centroid feature selection in text classification with high efficiency of calculation proven on English text document and now they are proven on Vietnamese text document. There are many classification techniques, but we implemented on the learning machine algorithms support vector machines. This method has been proven to be effective for text classification problems. With the technique of feature selection singular value decomposition and optimal orthogonal centroid feature selection, the implementing result higher than that of traditional method.

Keywords: *Text Mining , Feature Selection, Text Classification*

----------***----------

1. INTRODUCTION

Text mining [1] is a narrow research field of data mining, which focuses on discovering new information from text document collections, mainly by using techniques from data mining, machine learning, natural language processing and information retrieval. Text classification [2, 3] is content analysis task of the text and then giving decision whether this text belongs to which group among given text ones. Sentiment classification is a special kind of text classification in which a document is classified to predict automatically sentiment polarity (positive or negative). Sentiment is expressed usually in writing, it is showed on products blog posts, shopping sites, websites, forums, etc. Sentiment classification can be referred to as opinion mining as both study people's opinions, evaluation, feeling expression to products, organization, as well as to contents via events, attribute. It can be used in various fields, such as business intelligence, recommender systems, politics and psychology. Sentiment classification develops more and more and attracts the community's attention because it brings benefits to human's life and society. Automatic sentiment classification has been extensively studied and applied in recent years. However, sentiment is expressed differently in different domains, and annotating corpora for every possible domain of interest is impractical. We investigate domain adaptation for sentiment classifiers, focusing on online reviews for different types of products. Its application to many different domains of discourse makes it an ideal candidate for domain adaptation.

Many machine learning methods have been proposed for text categorization, so far, automatic text classification on the languages has enjoyed tremendous achievements, with most machine learning methods applied to the problem of text categorization which makes use of presentations of text in the form of feature vectors. The sole difference is the very selected feature space. In practice, words are in great numbers in a text, which results in thousands of features; in other words, each vector would have a large array of dimensionalities. The vectors are therefore of dissimilar sizes. To solve this problem, it is advisable to carry out such feature selections which are assessed as useful and ignore those of lesser importance. This process is very significant to text classification as a textual vector has a large number of dimensionalities, also with lots of redundancies. Feature selection methods prove to be very efficient in reducing the dimensionalities of textual feature vectors and yielding more effective text classification.

Feature selection also called subset selection or variable selection is a process which is generally applied in machine learning to solve the high dimensionality problem. Its duty is to select a subset of important features and to remove irrelevant, redundant and noisy features for simpler and more concise data representation. The benefits of feature

selection are multifold. Firstly, feature selection saves much of the running time of a learning process by removing irrelevant and redundant features. Secondly, irrelevant, redundant and noisy features do not interfere, learning algorithms can concentrate on the most important of data and construct simpler but more precise data models. Thus, the performance of classification is improved. Thirdly, feature selection can assist us to construct a simpler and more general model and to get a better insight into the underlying concept of the task [4-6]. Feature selection is a crucial problem for text classification and it determines the features relating most to the classification process. The reason is that some of the words are much more likely to be correlated to the class distribution than others. To use machine learning methods in text categorization, choosing the most suitable features to this task is important. In Text classification, algorithms of feature selection are a strategy to make text classifiers more efficient and accurate. Feature selection is a process of selecting a subset of features, in which the optimality of the subset is measured by an evaluation criterion. Feature selection attempts to cut down the dimension considered in a task to improve the performance on some dependent measures. Vocabulary reduction is often used to improve the categorization model performance. High dimensionality of the feature space can be an obstacle for using learning algorithms, reducing the feature space without compromising the classification model performance is crucial.

2. FEATURE AND FEATURE SELECTION PROCESS

2.1 The concept feature

The features of a text are the terms in the text. In representing the feature space during classification, feature selection or feature extraction is normally used [7]. The target of feature selection is to select a subset of representing features from the original feature space, while feature extraction is aimed at transforming the original feature space into a smaller feature space to reduce the feature size. Feature extraction cannot only reduce the feature size, but also succeed in solving the issue of polysemy and synonymy of words.

2.2 Feature Selection Process

After the processing are removed, the text can be seen as a set of features. Text classification will rely on these features. The number of features of the text is large and the space of features of all the texts being considered is very large, in principle, it includes all words in a language; therefore, features should be selected to shorten the dimensionality of the feature space. In fact, the process of feature selection is reducing the dimensionality of the feature vector by removing the unimportant feature components. In reality, we can not consider all words of a language but use a set of words drawn from a set (big enough) of the documents being considered (called the corpus).

In text classification tasks, all the features in the training dataset are independently evaluated by a specific feature selection function, a score is assigned to each single feature, and then the features are sorted according to their scores. Lastly, a predefined number of best features is selected to form a feature subset.

2.3. Feature weightenization

+ Weight
Weight is a typical value for the term, this value is usually real numbers. The frequently used formula is terms frequency inverse document frequency (TF_IDF) or its extension type logTF_IDF, terms frequency inverse word frequency (TF_IWF) [8].

+ The method of word weighted
There are many methods to calculate word weighting, however, the method of tf*idf weighting is applied in this study. This method balances the factors of recall level and feature number which are used in text document.

Given the set of feature $F=\{f_1, f_2, \ldots, f_N\}$ and set of classes $C=\{C_1, C_2, \ldots, C_N\}$, a feature $f \in F$ and a document d is weighted by the following formula:
+ Term frequency (TF): tf_{fd} is the occurrence frequency of feature f in document d. This weighting method states that a word is important for a document if it appears more than once in the document.

+ Terms frequency inverse document frequency (TF_IDF) is the most common scheme for weighting features, is calculated by the following formula:

$$w_{fd} = \frac{tf_{fd} \log(N / df_f)}{\sqrt{\sum_{r \in F} (f_{rd})^2 [\log(N / df_r)]^2}} \quad (2.1)$$

Where:
- w_{fd} the feature weight f in document d.
- tf_{fd} the occurrence frequency of feature f in document d.
- N the total of documents in the training set
- df_f the number of documents containing feature f.

+ logTF_IDF: is calculated by the following formula:

$$w_f^d = \frac{l_f^d \log(N / df_f)}{\sqrt{\sum_{r \in F} (l_r^d \log(N / df_r)^2}} \quad (2.2)$$

With:

$l_f^d = 0$ if $tf_{fd} = 0$ vice versa $l_f^d = \log(tf_{fd}) + 1$ if $tf_{fd} \neq 0$

+ TF_IWF is calculated by the following formula:

$$w_f^d = \frac{tf_{df} IWF(f)^2}{\sqrt{\sum o_r^d [IWF(r)^2]^2}}$$

(2.3)
With:

$$IWF = \log(\frac{O}{O_f})$$

Where:
- O the occurrence frequency of all features.
- O_f the occurrence frequency of feature f.

In fact, words have the different important levels in text document and in the classification. Many conjunction and numeral words have indistinguishability in the processing of classification. Apart from this, many other words which are unvaluable in classifying as words appear in many text documents or words rarely appear. Those words are called stop words and it is essential to remove those words in the classification. Here we choose TF_IDF and logTF_IDF to calculate in the experiment.

3. METHODOLOGY

We use 2 different methods: singular value decomposition (SVD) [9] and optimal orthogonal centroid feature selection (OCFS) for experiment [10]. These methods have been proved by experiment on the standard corpus of the world, considering English as the best. So the feasibility of this method is very high.

3.1 Optimal Orthogonal Centroid Feature Selection

The method OCFS is suggested in 2004 by authors of the Microsoft Asia research centre and is evaluated better than the previuos methods. OCFS method is based on the Orthogonal Centroid (OC) algorithm.

+ Orthogonal Centroid Algorithm

Orthogonal Centroid algorithm [10] is a proposed supervised feature extraction algorithm which utilizes orthogonal transformation on centroid. It has been proved very effective for classification problems on text data. OC algorithm is based on the Vector Space Computation in linear algebra by QR matrix decomposition. The OC algorithm also aims to find the transformation matrix $W \in R^{dxp}$ that maps each column $x_i \in R^d$ of $X \in R^{dxp}$ to a vector $y_i \in R^p$, Criterion $J(\tilde{W})$ is arg max J(W) = arg max trace($W^T S_b W$), subject to $W^T W = I$

$$S_b = \sum_{j=1}^{c} \frac{n_j}{n}(m_j - m)(m_j - m)^T$$

where:
- m_j the mean vector of the j^{th} class is $m_j = (1/n_j)\sum_{x_i \in c_j} x_i$
- n_j is the class size of j
- m the mean vector of all these documents is

$$m = (1/n)\sum_{i=1}^{n} x_i = (1/n)\sum_{j=1}^{c} n_j m_i$$

+ OCFS Algorithm [10]

The solution space of the feature selection problem is discrete and consists of all matrices $W \in R^{dxp}$ that satisfy the constraint given above. the feature selection problem according to criterion $J(\tilde{W})$ is an optimization problem: $\arg\max J(\tilde{W}) = \arg\max trace(\tilde{W}^T S_b \tilde{W})$ subject to $\tilde{W} \in H^{dxp}$.

The elements in the formula are defined as in the OC section. Suppose $K = \{k_i, 1 \le k_i \le d, i = 1,2,..., p\}$ is a group of indices of features. $\tilde{W}$ is a binary matrix with its elements of 0 or 1, and there are 1 and only 1 no 0 element in each column. On the other hand, we have:

$$trace(\tilde{W}^T S_b \tilde{W}) = \sum_{i=1}^{p} \tilde{w}_i^T S_b \tilde{w}_i = \sum_{i=1}^{p}\sum_{j=1}^{c} \frac{n_j}{n}(m_j^{k_i} - m^{k_i})^2 \quad (3.1)$$

Thus, the nature of the OCFS issue is to look for the above set K as a maximum:

$$\sum_{i=1}^{p}\sum_{j=1}^{c} \frac{n_j}{n}(m_j^{k_i} - m^{k_i})^2 \quad (3.2)$$

This motivates to propose an optimal feature selection algorithm according to $J(\tilde{W})$.

The details of the OCFS algorithm:
+ Input: Datasets (Copus)
+ Method:
- Step 1, compute the centroid m_i, i=1, 2,…, c of each class for training data;
- Step 2, compute the centroid m of all training samples;
- Step 3, compute feature score

$$s(i) = \sum_{j=1}^{c} \frac{n_j}{n}(m_j^i - m^i)^2$$

for all the features;
- Step 4, find the corresponding index set K consisted of the p largest ones in set S={s(i)| 1≤i ≤ d}.

+ Out put: Recall Precision and Micro F1 of datasets.
+ Algorithm Analysis
- In terms of calculation: The OCFS algorithm complexity is O(cd). The OCFS algorithm is easy to be installed, faster in calculation and more effective than the 5 methods mentioned above.

- In terms of number of selected features: Without loss of generality, suppose the feature score of all the d features are s(k1) ≥ s(k2) ≥…≥ s(kd), the energy function is defined as:

$$E(p) = \frac{\sum_{j=1}^{p} s(k_j)}{\sum_{i=1}^{d} s(i)}$$

we can get the optimal number of features p* by:
p* = arg min E(p) subject to E(p) ≥ T, where giving a threshold such as T=80%.

3.2 Singular Value Decomposition Method

Singular value decomposition method [9] is the basic mathematical decomposition in technique indicating latent semantic indexing, which is commonly used in searching and retrieving information under text document form. The main idea of the algorithm is as follows: For a given matrix A (mxn size), matrix A is decomposed of the multiplication of 3 matrices as the following form:

$A = U \Sigma V^T$, of which:
- U is the mxm orthogonal matrix having the left singular vectors of A as its columns.
- V is the nxn orthogonal matrix having the right singular vectors of A as its columns.

- Σ is the mxn diagonal matrix having the singular values, not negative and the order descending:
$\delta 1 \geq \delta 2 \geq \cdots \geq \delta \min(m,n) \geq 0$.

Rank of matrix A is equal to the number of nonzero singular values. Typically, A is a sparse matrix with large size. To reduce the dimensional number of the matrix, normally, matrix A (with rank r) is approximated as a matrix A_k with a much smaller rank than r. The approximation matrix of A with this technique is $A_k = U_k \Sigma_k V_k^T$, of which:
- U_k is the orthogonal matrix mxk with columns k, first columns of matrix U.

- Σ_k is the diagonal matrix kxk containing first characters δ_1 , δ_2,…δ_k on the main diagonal.

- V_k is orthogonal matrix nxk with columns k, first columns of matrix V.

This approximation is considered as converting the current space (r dimension) into k dimensional space where k is much smaller than r.

In the practice, cutting dimension k of matrix A removes disturbance and enhances links of latent semantic among words in the text document files. In the study, the approximation technique is applied to shorten the dimensional number of the feature space.

Firstly, each text document is modeled into a column vector in the space identified by A_{mxn}. After removing A_{mxn} to A_k, all the current vectors are showed on the space A_k to get the dimensional number k as the formula:

$$Proj(x) = x^T U_k \sum_k^{-1}$$

4. EXPERIMENTAL RESULTS

In this part, the implementation process and evaluation with technique of feature selection OCFS and SVD, the method suggested in part 3 are presented. In the meantime, the classic method Mutual Information (MI) [10] is used as a basic to compare it to our recommended method.

4.1 Evaluation Method

Definition of the measure values:

+ Recall (R): $$R = \frac{N_1}{T}$$

+ Precision (P): $$P = \frac{N_2}{N_3}$$

+ Micro F1: $$F1 = \frac{2PR}{P+R}$$

Where:
- P is the Precision; R: is the Recall; F1 to denote Micro F1
- N_1: total number of documents in the training set yielding correct test results by the model.
- T: total number of documents of training sets.
- N_2: total number of documents in the test set yielding correct test results by the model.
- N_3: total number of documents in the test set

4.2 Results

We implemented on the learning machine algorithms SVM (Support Vector Machines) by using library LIBSVM [11]. This method has been proven to be effective for text classification problems. The sample document booklet includes 2868 documents retrieved from the webpage http://www.nguoitieudung.com.vn/ (table 1). This data is divided into 2 parts: 50% used for training document and 50% for test document.

Table 1. 11 topics and amount of samples used in the test

Name topic	Number of training samples	Number of test samples	Total sample texts
Informatics	106	106	212
Lifestyle	97	97	194
Economy	73	73	146
Fashion	234	234	468
Law	213	213	426
Education	155	155	310
Products and services	124	124	248
Health	144	144	288
Eating and drinking	62	62	124
Culture and recreation	141	141	282
Tourism	85	85	170

Text data are processed by separating sections, sentences, and standardizing the spelling and delete stopwords . after modeling, each document is a weight vector of words. The formula to calculate the value is TF_IDF and logTF_IDF [6] in section 3. Experimental results on the two methods are as follow (table 2).

(5.26)

Table 2. Experimental result table

Method	Recall	Precision	F1
OCFS	93.21%	90.08%	91.62%
MI	80.78%	77.87%	78.70%
SVD	96.41%	92.73%	94.53%

(5.27)

(5.28)

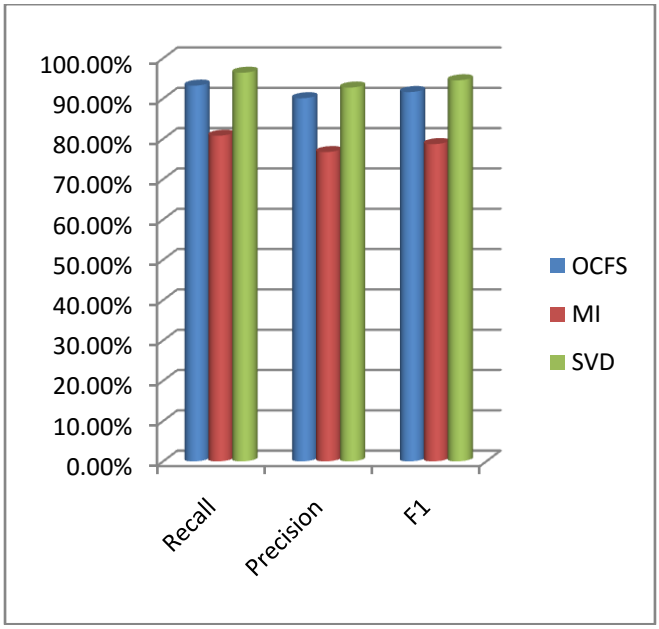

Chart -1: Experimental result of three method

Thus, with the technique of feature selection OCFS, the implementing result is 91.62% higher than that of traditional method MI of 78.70%. The result on the technique SVD is 94.53% higher than that on the method OCFS. These rerult reconfirm the effectiveness that we have done on English texts.

5. CONCLUSIONS

Text classification problem is the one with many featues, thus to improve the effectiveness of text classification is the aim of may researchers. Shortening the dimensions of text feature space by the method optimal orthogonal centroid feature selection and singular value decomposition in text classification is proven to be sucessful on the problems with big feature space like Vietnamese text classification problem. Feature selection plays an important role in the process of classification and impacts on the classifying effectiveness, thus it is worth considering to decide which method to use for better implementing result. In this study, we present two methods of feature selection in text classification with high efficiency of calculation proven on English text document; and now they are proven on Vietnamese text document.

REFERENCES

[1]. Han, J.; Kamber, M. Data Mining: Concepts and Techniques. Morgan Kaufmann, Burlington, US, 2006
[2]. Y. Song, J. Huang, D. Zhou, H. Zha, and C. L. Giles. IKNN: Informative k-nearest neighbor pattern classification. In PKDD 2007, pages 248–264.
[3]. Y. Song, D. Zhou, J. Huang, I. G. Councill, H. Zha, and C. L. Giles. Boosting the feature space: Text classification for unstructured data on the web. In ICDM ’06: Proceedings of the Sixth International Conference on Data Mining, Washington, DC, USA. IEEE Computer Society, 2006, pages 1064–1069.
[4]. Dash, M. and Liu, H. Feature selection for classification. International Journal of Intelligent Data Analysis, 1(3), 1997.
[5]. Kohavi, R. and John, G. H. Wrappers for Feature Subset Selection. Artificial Intelligence, vol. 97, nos. 1-2, 1997, pp. 273-324.
[6]. Koller, D. and Sahami, M. Toward optimal feature selection. In: Proceedings of International Conference on Machine Learning, 1996.
[7]. Tao Liu, Zheng Chen, Benyu Zhang, Wei-ying Ma, Gongti Wu. Improving Text Classificaion using Local Latent Semantic Indexing, Data Mining, 2004. ICDM 2004. Proceedings, Fourth IEEE International Conference. 2004
[8]. Yang, Y. and Pedersen , J,O. , A comparative Study On Feature Selection in Text Categorization. In Proceedings of the 14th International Conference on Machine Learning(ICML), (1997), 412-420.
[9]. Golub, G.H., Loan, C.F.V., 1996. Matrix Computations, third ed. Johns Hopkins University Press, pp. 48–80
[10]. Jun Yan-Ning Liu-Benyu Zhang-Shuicheng Yan. (2005) OCFS: Optimal Orthogonal Centroid Feature Selection for Text Categorization, Microsoft Research Asia, China.
[11]. http://www.csie.ntu.edu.tw/~cjlin/libsvm/

BIOGRAPHIES

Description about the author
Full name: **Hoan Dau Manh**
Gender: Male.
University studies: Ph. D. degree in computer science and technology.
Experience: more than 15 years of research and teaching.
Scientific interest: Data base, data mining, machine learning and natural language processing