I.J. Information Engineering and Electronic Business, **2012, 1, 44-50**
Published Online February 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijieeb.2012.01.06

A Study on Analysis of SMS Classification Using Document Frequency Thresold

R.*Parimala* (Corresponding author)
Research Scholar, National Institute of Technology
Tiruchirappalli
E-mail: rajamohanparimala@gmail.com

Dr. R. Nallaswamy
Professor, Department of Mathematics
National Institute of Technology, Tiruchirappalli
E-mail: nalla@nitt.edu

***Abstract*—Recent years, feature selection is chief concern in text classification. A major characteristic in text classification is the high dimensionality of the feature space. Therefore, feature selection is strongly considered as one of the crucial part in text document categorization. Selecting the best features to represent documents can reduce the dimensionality of feature space hence increase the performance. Feature selection is performed here using Document Frequency Threshold. This paper focus on SVM based text message classification using document frequency threshold. The experiment is performed with NUS SMS text messages data set. An experimental result shows that the results of proposed method are more efficient.**

***Index Terms*—Text Mining, Support Vector Machine, Document Term Matrix, Document frequency threshold.**

1. Introduction

SMS spam (sometimes called cell phone spam) is any junk message delivered to a mobile phone as text messaging through the Short Message Service (SMS). Although SMS spam is less prevalent than email spam, it still accounts for roughly 1% of texts sent in the United States and 30% of text messages sent in parts of Asia. In many western countries, mobile subscribers view unsolicited messages via SMS as an intrusion of their privacy. Receiving unsolicited and potentially malicious messages often incenses subscribers, compelling them to call their MNO to complain. Nowadays, SMS spam a major problem in India, has become the world's largest and fastest-growing mobile market with over 700 million subscribers. Many people expect only the most urgent of messages on their cellular phones. From the survey of user perceptions about SMS spam, it has been found that perception of more than one user about the same SMS may differ. In recent times, there are many media reports published on SMS spam problem [1][2]. The most important fact here is that end users are helpless in controlling the number of SMS spam they are receiving. Gomez Hidalgo et al, have previously reported the effect of feature engineering on the application of standard classifiers to SMS messages

Text message classification is the classification of messages with respect to a set of one or more predefined categories. Although many approaches proposed, text categorization is still a major area of research primarily because the effectiveness of current text classifiers is not faultless and still needs improvement. . With all the effort in this domain, there is still a place for improvement and a great deal of attention is paid to developing highly accurate classifiers with less computational cost. The most common text representation is Bag of word approach (BoW). Here text is represented as a vector. The BoW vectors are then refined by feature selection, where vectors are removed from the representation using computationally less feature selection measure, Document Frequency threshold. The reduced feature set fed to the Support Vector Machine(SVM) classifier to classify the text messages. .

The rest of the paper is organized as follows: Section 2 gives Introduction to Support Vector Machine. Section 3 presents a detail about NUS SMS Text Collection; Used Environment and Libraries and Performance measure Section 4 describes proposed method. Section 5 presents Results. Finally Section 5 concludes the paper.

2. Introduction to Support Vector Machine

2.1 An Overview of Classifier

Machine learning methods, including Support Vector Machines (SVMs), have tremendous potential for helping people more effectively and organize electronic resources. As a powerful statistical model with ability to handle a very large feature set, SVM is widely used in pattern recognition areas such as face detection, isolated handwriting digit recognition, and gene classification [3]. Recently SVM has been used for text categorization successfully. T. Joachims [4] classified documents into categories by using SVM and obtained better results than those obtained by using other machine learning techniques such as Bayes and K-NN. Similarly, J.T. Kwok [5] used SVM, to classify Reuters newswire stories into categories and obtained better results than using a k-NN classifier.

Support Vector Machines (SVM) is a machine learning model proposed by V. N. Vapnik [6]. The basic idea of SVM is to find an optimal hyperplane to separate two classes with the largest margin from pre-classified data. After this hyperplane is determined, it is used for classifying data into two classes based on which side they are located. By applying appropriate transformations to the data space before computing the separating hyperplane, SVM can be extended to cases where the margin between two classes is non-linear.

2.2 Maximal Margin Hyperplane

Text classification is usually achieved by using Machine Learning techniques, which acquire of labeled documents and no human intervention for coding rules or heuristics. Machine Learning algorithm has generated a model of the training data, used to classify new un-labeled documents automatically. SVM is a new paradigm of learning system. Since 1990s, SVM has been a promising tool for data classification. This introduction to Support Vector Machines (SVMs) is based on [3], [6], [7] and [8]. Support vector machines (SVMs) [6] are of great interest to theoretical and applied researchers and they have strong connections to computational learning theory. The basic idea is easiest to understand, when we have a linearly separable two-class problem. The resulting classifier called the maximal margin classifier. The idea is to search the optimal separating hyper plane which has the maximal margin of separation between the training vectors from the two classes, so maximal margin classifiers estimate directly the decision boundary. A separating hyperplane means that the training vectors from the two classes lie on different sides of the hyperplane, and having maximal margin means that distance from the hyperplane to the nearest training vector is maximal. The support vectors are those training vectors which lie nearest to the optimal hyperplane. This optimization problem formulated as a quadratic programming problem. In real applications, the training data is usually not linearly separable and then the maximal margin hyperplane does not exist. A solution is to seek the so-called soft-margin hyperplane instead. Also this leads to a quadratic program. Since interpret of SVM classifiers leads to standard convex optimization problems, no complications with local minima as there are with MLPs. These quadratic programs solved either by general purpose quadratic program solvers or by techniques developed specially for SVMs.

If the training data are linearly separable, then there exists a pair $(\mathbf{w}, b)$ such that

$$\begin{aligned} \mathbf{w}^T\mathbf{x}_i + b &\geq 1, \text{ for all } \mathbf{x}_i \in P \\ \mathbf{w}^T\mathbf{x}_i + b &\leq -1, \text{ for all } \mathbf{x}_i \in N \end{aligned} \qquad (1)$$

The decision function is of the form

$$f(x) = \operatorname{sgn}(w^t x + b). \qquad (2)$$

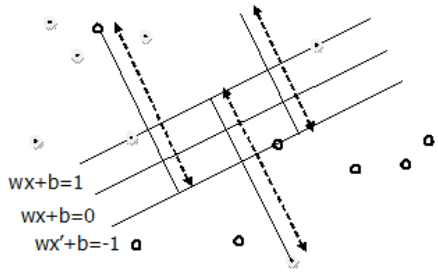


Fig1. Optimal separating hyperplane for Binary classification problem.

$\mathbf{w}$ is termed the weight vector and b the bias (or $-b$ is termed the threshold). The inequality constraints (1) can be combined to give

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \text{for all } \mathbf{x}_i \in P \cup N \qquad (1)$$

2.3 Support Vector Machines

Given a training set of instance-label pairs (x_i, y_i), i =1, 2, 3, …ℓ where $x_i \in R^n$, the class label of the ith pattern is denoted by $y_i \in \{1,-1\}^t$. Nonlinearly separable problem are often solved by mapping the input data samples x_i to a higher dimensional feature space $\phi(x_i)$. The classical maximum margin SVM classifier aims to find a hyperplane of the form $w^t\phi(x) + b = 0$, which separates patterns of the two classes. So far we have restricted ourselves to the case where the two classes are noise-free. In the case of noisy data, forcing zero training error will lead to poor generalization. To take account of the fact that some data points misclassified, we introduce a vector of slack variables $\Xi = (\xi_1, \ldots, \xi_l)^T$ that measure the amount of violation of the constraints (3). The problem can then be written as

$$\underset{w,b,\xi}{Minimize} \frac{1}{2} w^t w + C \sum_{i=1}^{n} \xi_i$$

(4) subject to the constraints

$$y_i\left(w^t\phi(x_i) + b\right) \geq 1 - \xi_i$$
$$\xi_i \geq 0, i = 1, 2, 3 \ldots\ldots \ell,$$

(5)

The solution to (4)-(5) yields the soft margin classifier, is termed because the distance or margin between the separating hyperplane $w^t\left(\phi(x) + b\right) = 0$ is usually determined by considering the dual problem, which is given by

$$L(\mathbf{w}, b, \alpha_i, \Xi, \Gamma) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{l} \alpha_i [y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) - 1 + \xi_i] - \sum_{i=1}^{l} \gamma_i \xi_i + C\sum_{i=1}^{l} \xi_i$$

where $\Lambda = (\alpha_1, \ldots, \alpha_l)^T$ and $\Gamma = (\gamma_1, \ldots, \gamma_l)^T$, are the Lagrange multipliers corresponding to the positivity of the slack variables. The solution of this problem is the saddle point of the Lagrangian given by minimizing L with respect to $\mathbf{w}, \Xi$ and b , and maximizing with respect to $\Lambda \geq 0$ and $\Gamma \geq 0$. Differentiating with respect to w, b and Ξ and setting the results equal to zero,

It is obtained

$$\frac{\partial L(\mathbf{w}, b, \alpha, \Xi, \Gamma)}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{l} \alpha_i y_i \phi(\mathbf{x}_i) = 0,$$

$$\frac{\partial L(\mathbf{w}, b, \alpha, \Xi, \Gamma)}{\partial b} = -\sum_{i=1}^{l} \alpha_i y_i = 0 ,$$

and

$$\frac{\partial L(\mathbf{w}, b, \Lambda, \Xi, \Gamma)}{\partial \xi_i} = C - \alpha_i - \gamma_i = 0.$$

$$\underset{\alpha}{Minimize} \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j k\left(x_i, x_j\right) - \sum_{i=1}^{\ell} \alpha_i$$

subject to

$$\sum_{i=1}^{\ell} \alpha_i y_i = 0, \text{ and } 0 \le \alpha_i \le C, i = 1,2,3.....\ell \quad (7)$$

Here, $\alpha_{i,} i = 1,2,3,....\ell$ denotes the Lagrange multipliers and the matrix $K\left(x_i, x_j\right) = \phi\left(x_i\right).\phi\left(x_j\right)$ are termed as Kernel matrix [9][10]. Training vector x_i is mapped into a higher dimensional feature space and construct an optimal hyperplane. SVM also restricts the choice of Kernel that the Quadratic programming is a convex problem. Therefore, it guarantees that global optimization with corresponding Kernel. SVM uses training data as Support Vectors and uses Lagrange multipliers to represent the Support Vectors. The classifier can be constructed using the decision function in the form

$$y(x) = sign\left[\sum_{k=1}^{\ell} \alpha_k y_k K(x, x_k) + b\right].$$

3.1 SMS Text Collection

A collection of about 10,000 legitimate messages collected for research at the Department of Computer Science at the National University of Singapore, called the *NUS SMS Corpus* (NSC). A collection of English SMS messages, including 1002 legitimate messages randomly extracted from the NUS SMS Corpus and the Jon Stevenson Corpus, and 82 SMS spam messages collected from the Grumbletext mobile spam site. This is a UK forum in which cell phone users make public claims about SMS spam messages, most of them without reporting the very spam message received. The identification of the text of spam messages in the claims is a very hard and time-consuming task, and it involved carefully scanning 100 messages. It is believed this collection resembles a realistic scenario, because both the legitimate and the spam messages are real messages; the proportion may be not accurate but we are not aware of the existence of real world statistics of spam received by cell phone users in the British/Singapore markets.

3.2 Used Environment and Libraries

Within the past few years tm has gained interest from a variety of researchers and users of different backgrounds [12], [13]. **R** is a programming language and software environment for statistical computing and graphics. R is more than a programming language. It is an interactive environment for doing statistics. The R language is the scripting language for the R environment. An R interface has been added to the popular data mining software Weka which allows for the use of the data mining capabilities in Weka and statistical analysis in R. kernlab for kernel learning provides ksvm and is more integrated into R so that different kernels can easily be explored [14],[15]. The machine used was an Intel Core 2 Duo E7500 @ 2.93GHz with 2GB RAM.

3.3 Performance Measure

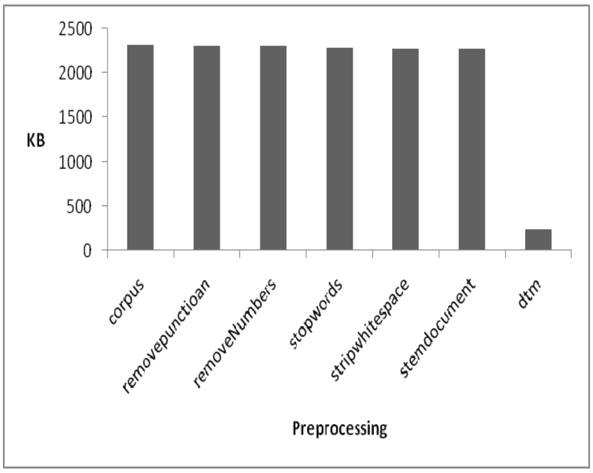
The research intends to compare the efficiency of SVM. Detection and identification of spam and non-spam SMS generalized as the following: True positive (TP): the number of spam SMS detected when it is actually spam SMS. True negative (TN): the number of non-spam SMS detected when it is actually non-spam. Classifiers have long been evaluated on their accuracy only. An often-used measure in the information retrieval and natural language processing communities is Overall Accuracy. An often-used another measure in the information retrieval and natural language processing communities is the F1-measure. According to Yang and Liu [16], this measure was first introduced by C. J. Van Rijsbergen [17]. They state, the F1 measure combines recall (R) and precision (P) with an equal weight in the following form: $\frac{2RP}{R + P}$, where $R = \frac{TP}{TP + FN} x100\,\%$ and $P = \frac{TP}{FP + TP} x100\,\%$. TP is the number of true positives, i.e., the number of non-spam SMS cases predicted correctly. TN is the number of true negatives, i.e., the number of cases correctly predicted as non-spam. FP is the number of false positives, i.e., the number of cases incorrectly predicted as non-spam. FN is the number of false negatives, i.e., the number of cases incorrectly predicted as spam.

4. Proposed Method

4.1 Preprocessing

The first step of text mining process is text preprocessing in which the document collection is analyzed syntactically or semantically. The text message document is considered as a bag of words because the words and its occurrences are used to represent the document. The algorithms applied at this stage are stemming and stop word removal, number removal and strip whitespaces. The next tasks for text mining include creating a Document Term Matrix (DTM), identifying frequently occurring words, and removing sparse terms. Tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation. Stop words are those common words that do not add meaningful content to the document (auxiliary verbs, conjunctions and articles). Stemming reduces the number of unique words. Term frequency is the frequency of occurrence of a term in a particular document collection or a query collection. The text size of text preprocessing is summarized in fig2.

Fig 2. Text preprocessing



4.2 Direct use of SVM

After preprocessing, two-thirds of the data set is used as training set for SVM and one-third is used to measure the classification accuracy of the SVM. The R software kernlab is employed to solve the quadratic programming problem. The radial basis function is adopted as the kernel function, and 10-fold cross validation is performed to get the nice arguments received the result to train model and do the final prediction [11].

4.3. Document Frequency Threshold

Document frequency of a feature is the number of documents in which the frequency occurs. Document frequency is class independent because of its simple computation and good performance, Document frequency has been widely used in Dimensionality reduction. Usually, the term that rarely appear in the Corpus provide little specific information and do not affect the global prediction performance. Most frequent and rare frequent terms are removed from the data set and then classification is performed.

5. Results

The sample features are given as

```
> s$account
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0
0
```

```
> s$articles
 [1] 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0
> s$articles
 [1] 0 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0
> s$clean
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0
0
> s$click
 [1]  0  0  0  0  0  0  0  0  0  0  0  0  0  0 17  0  0  0  0  0  0
0  0  0  0
[26]  0  0  0  0
>
```

It is observed that the terms of medium frequency have usually high semantic content. This method has a time complexity O(f), where f is the number of features.

The performance measure of direct use of SVM is shown in Table 1.

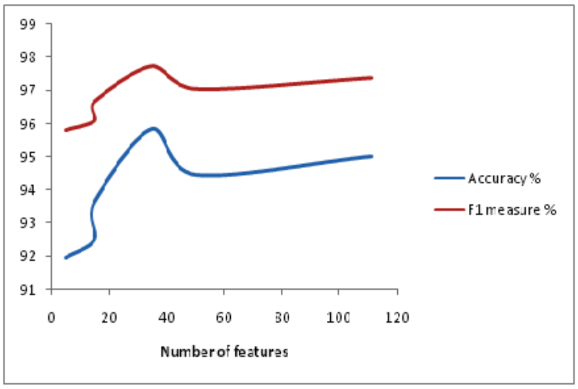
Table.1 Performance Measure of SVM before Document Frequency Threshold

Corpus Size (Kb)	Sparsity %	No. of features	DTM Size (Kb)	SVM Acc %	SVM F1 Measure
2311.4	99.5	262	118.9	97.77	92.41

Table.2 Performance Measure of SVM after Document Frequency Threshold

Frequent Term Range	No .of features	SVM Accuracy %	SVM F1 measure %
5-10	111	95.00	97.37
10-15	50	94.44	97.04
15-20	35	95.83	97.74
20-25	15	93.61	96.65
25-30	15	92.50	96.09
30-35	5	91.94	95.80

Fig3. Performance Measure NUS SMS Data After Document frequency threshold



5. Conclusion

In this paper, a document frequency feature selection criteria for SMS classification based on support vector machine classifier is proposed. The document threshold is class independent. Select features without using any statistical measure such as information gain, correlation etc., and this method simply select the features without any prior knowledge about features. The proposed medium Document frequency threshold is used to enhance the F1-measure with less computational cost. This method have been implemented in R Programming language

References

[1]http://timesofindiaindiatimes.com/tech/personaltech/computing/junk-sms-no-end-tomobile pammes/articleshow/6247207 .cms.

[2]http://www.livemint.com/2010/07/27000020/Scour e-of-SMS-spam-swamps-mob.html.

[3] C.J.C. Burges., A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2): 955-974, 1998.

[4] T. Joachims, Learning to Classify Text Using Support Vector Machines Dissertation, Kluwer, 2002.

[5] J.T. Kwok. Automated text categorization using support vector machine, In Proceedings of the International Conference on Neural Information Processing, Kitakyushu, Japan, Oct. 1998, pp. 347-351.

[6] V. N. Vapnik. The nature of Statistical Learning Theory, Springer, Berlin, 1995.

[7] N. Cristianini, and J. Shawe-Taylor, Support Vector and Kernel Methods, Intelligent Data Analysis: An Introduction Springer – Verlag, 2003.

[8] N.Cristianini, and J. Shawe-Taylor, An introduction to support vector machines, Cambridge, UK: Cambridge University Press, 2004.

[9] B. Schölkopf. C.J.C. Burges, and A.J. Smola,Advances in Kernel Methods: Support Vector Learning, MIT Press, (Eds.), 1998.

[10] A.J. Smola and B. Scholkopf, Learning with kernels: Support Vector Machines, regularization, optimization, and beyond, Cambridge, MA: MIT press.

[11] SU Gao-li, Deng Fang-ping. Introduction to Model selection of SVM Regression, Bulletin of Science and Technology, 2006.22(2):154-157

[12] Ingo Feinerer. An introduction to text mining in R. *R News*, 8(2):19-22, October 2008

[13] Ingo Feinerer, Kurt Hornik, and David Meyer. Text mining infrastructure in R. Journal of Statistical Software, 25(5):1-54, March 2008.

[14] Karatzoglou, A., Smola, A., Hornik, K,, Zeileis, A., 2005, kernlab–Kernel Methods., R package, Version 0.6-2., Available from http://cran.R-project.org.

[15] Alexandros Karatzoglou and Ingo. Feinerer. Kernel-based machine learning for fast text mining in R. Computational Statistics & Data Analysis, 54(2):290-297, February 2010.

[16] Y. Yang and X. Liu., 1999, A re-examination of text categorization methods, In Proc of SIGIR, ACM press, NewYork, NY, USA.

[17] C. J. van Rijsbergen., 1979, Information Retrieval. Butterworth's, London.

R. Parimala: Assistant professor of Department of computer science, Periyar, E.V.R college, Tiruchirapalli, interested in soft computing, data mining and optimization techniques.

Dr. R. Nallaswamy: Professor of Department of Mathematics in National Institute of Technology, Tiruchirapalli, interested in bio-mathematics, optimization techniques and soft computing.