Journal of Intelligent Learning Systems and Applications, **2013, 5, 84-89**
http://dx.doi.org/10.4236/jilsa.2013.52009 Published Online May 2013 (http://www.scirp.org/journal/jilsa)

The Role of Rare Terms in Enhancing the Performance of Polynomial Networks Based Text Categorization

Mayy M. Al-Tahrawi

Department of Computer Information Systems, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan.
Email: mtahrawi@ammanu.edu.jo

Received March 13th, 2013; revised May 2nd, 2013; accepted May 9th, 2013

ABSTRACT

In this paper, the role of rare or infrequent terms in enhancing the accuracy of English Text Categorization using Polynomial Networks (PNs) is investigated. To study the impact of rare terms in enhancing the accuracy of PNs-based text categorization, different term reduction criteria as well as different term weighting schemes were experimented on the Reuters Corpus using PNs. Each term weighting scheme on each reduced term set was tested once keeping the rare terms and another time removing them. All the experiments conducted in this research show that keeping rare terms substantially improves the performance of Polynomial Networks in Text Categorization, regardless of the term reduction method, the number of terms used in classification, or the term weighting scheme adopted.

Keywords: Polynomial Networks; Text Categorization; Document Classification; Infrequent Terms; Rare Terms

1. Introduction

Text categorization (TC) or Document Classification is the task of automatically assigning an unseen document to one or more pre-defined categories, classes, or topics. The need for automatic, fast, accurate, and efficient classification of the huge amounts of online textual information grows rapidly, as the manual categorization of such huge information is not feasible considering time constraints, accuracy and cost of human trained professionals to perform this task. Polynomial Networks (PN) classifiers have proved to be competitive to the top performers in the field of English Text Categorization. They were compared with Support Vector Machines (SVM), Logistic Regression (LR), k-nearest-neighbor (kNN), Naive Bayes (NB), and the Radial Basis Function networks (RBF) on Reuters and 20Newsgroups and have achieved competitive performance non-iteratively, without the need for fine parameter tuning, and using just 0.25% - 0.5% of the corpora terms [1]. In text categorization, term selection is typically used to achieve two objectives: reducing the size of the term set in order to optimize the usage of computing resources and removing noise from the data in order to optimize the classification performance. Terms are often scored and ranked using some term weighting scheme that reflects the importance of the term for a given task. Only a selected subset of top scoring terms is used for further processing. The effect of rare or infrequent terms in the classification performance was always debatable. These terms were found to add noise in text categorization [2], while they were considered very helpful in improving the accuracy of text categorization in [3-6]. In fact, the inverse document frequency (idf), a famous term weighting scheme, is based essentially on the assumption that rare terms are no less important than frequent terms , and they were proved to be valuable in improving precision of text categorization [4,5]. The authors in [6] had shown, in their work, how rare terms can be used to improve classification accuracy. They found that rare words were able to indicate surprisingly well if two documents belong to the same category, and thus can aid classification and clustering. They also found that 5% - 25% of the test set can be classified essentially for free based on rare terms without any loss of accuracy. They even experienced an accuracy improvement of 0.6% - 1.6% when keeping rare terms. To investigate the role of the infrequent or rare terms in enhancing TC performance using Polynomial Networks (PNs), experiments were conducted, in this research, using the same term weighting scheme, dataset size and term reduction method, once keeping terms which occur just once in the documents, and another time discarding these terms. Three term weighting schemes and four term reduction methods were experimented. All the experiments

have shown that keeping infrequent terms has improved the PN classifier performance substantially regardless of the term set collection or the term weighting scheme, used in classification. The rest of the paper is organized as follows. Section 2 presents an overview of the PN classifier, while Section 3 is devoted to explain, in brief, the dataset used and the processing steps performed on the dataset. The term reduction (Term Selection) methods are presented in Section 4, while the term weighting schemes, used in the experiments, are covered in Section 5. The performance evaluation measures used in the experiments are presented in Section 6, and Section 7 of the paper presents a summary of the results reached in the experiments conducted in this research, while analysis of these results takes place in Section 8. Finally, conclusions and intended future work are presented in Section 9.

2. Polynomial Networks (PNs)

Polynomial Network (PN) classifiers have been known in the literature for many years [7], and have been recently used in some areas like speaker verification and sign language recognition [8-12]. More recently, PNs have proved to be able to achieve high text categorization accuracy, using just a very small subset of the terms of the two benchmark datasets in TC: Reuters and 20Newsgroups [1].

2.1. The Architecture of PNs

The adopted PN model consists of two layers. The first layer (the input layer) forms the monomial basis terms of the input vector $x(x_1, x_2, \cdots, x_N)$, such as $1, x_1, x_2, x_1^2, \cdots$, etc., where N is the number of terms (dimensions) of x. A second layer then linearly combines the output of the first layer; *i.e.* the data is first expanded into a high dimensional space in the first layer and then it is linearly separated using the second layer. The basic embodiment of a K^{th} order polynomial network consists of several parts. The N terms of one observation $x(x_1, x_2, \cdots, x_N)$ are used to form a basis function $p(x)$; one $p(x)$ is formed for each observation. The elements of $p(x)$ for a polynomial of degree K are monomials of the form [9]:

$$\prod_{j=1}^{N} x_j^{k_j}, \text{ where } k_j \geq 0 \text{ and } 0 \leq \sum_{j=1}^{N} k_j \leq K \quad (1)$$

The second layer of the PN linearly combines all inputs to produce weights of classes (classes' models). The whole class is represented by one weight, which is computed during the training phase. Detailed training steps are presented in the next section.

2.2. The Training Phase

A PN is trained to approximate an ideal output using mean squared error as the objective criterion. The polynomial expansion of the i^{th} class term vectors is denoted by [8]:

$$M_i = \left[p(x_i, 1), p(x_i, 2), p(x_i, 3), \cdots, p(x_i, N_i) \right]^t \quad (2)$$

where N_i is the number of training term vectors for class i, and $p(x_{i,m})$ is the basis function of the m^{th} term vector for class i. After forming M_i for each class i of the nc training classes, a global matrix M is obtained for the nc classes, by concatenating the individual M_i's computed for each class [9]

$$M = \left[M_1, M_2, M_3, \cdots, M_{nc} \right]^t \quad (3)$$

The training problem then reduces to finding an optimum set of weights w (one weight for each class) that minimizes the distance between the ideal outputs (targets) and a linear combination of the polynomial expansion of the training data such that [9]:

$$w_i^{opt} = \arg\min_{w} \left\| Mw - o_i \right\|_2 \quad (4)$$

where o_i is the ideal output (a column vector which contains N_i ones in the rows where the i^{th} class' data are located in M, and contains zeros otherwise). A class model w_i^{opt} can be obtained in one shot (non-iteratively) by applying the normal equations method [9,13]:

$$M^t M w_i^{opt} = M^t o_i \quad (5)$$

By defining MG as M^tM, Equation (7) reduces to

$$w_i^{opt} = MG^{-1} M^t o_i \quad (6)$$

2.3. Recognition

Recognition (classification of a new unseen document) consists of two parts: identification and verification. The identification phase proceeds as follows: The term vector x of the input (the unseen input to classify) is expanded into its polynomial terms $p(x)$ in a manner similar to what was done with the training inputs in the training phase (using the same polynomial degree). Then, the new unseen input is assigned to the class c such that [9]

$$c = \arg\max_{i} w_i^{opt} \cdot p(x) \text{ for } i = 1, 2, \cdots, nc \quad (7)$$

In the verification phase, classifications with scores above 0.5 were accepted, as the output score w_i. $p(x)$ lies between 0 and 1.

2.4. Text Categorization (TC) Using Polynomial Networks (PNs)

The training phase of TC using PNs goes through the following steps. Each training document is represented by a vector of terms x using the vector space model. Terms can be represented by their *binary* weights, *nor-*

malized frequencies, *term frequency-inverse document frequency* (*tf-idf*) weights, or any other weighting scheme. Then, the k^{th} order PN basis function $p(x)$ is formed for each training document, as in Equation (1). Second order PNs are used in the experiments presented in this paper. The polynomial expansion of each class i training files, M_i, is then formed as in Equation (2). Now, the global matrix for all the nc classes is obtained by concatenating all the individual M_i matrices into M as in Equation (3). Once the global matrix M is formed, the PN is trained to approximate an ideal output using mean-squared error as the objective criterion (Equation (4)). Finally, the training phase ends with finding the optimum set of weights w as in Equations (5) and (6). To classify an unseen document, the term vector x of the unseen document is expanded into its polynomial terms $p(x)$ as in Equation (1). Then, the new unseen document is assigned to class c as explained in Equation (7).

3. Data Set

The Reuters-21578 benchmark subset suitable for single-label text categorization R8 [14] was used in this research. The whole processing steps performed on the datasets can be summarized as follows:

1) Only letters, hyphens "-" and underscores "_" are kept; any other character is eliminated.

2) All letters are converted to lowercase.

3) Tabs, new lines, and RETURN characters are replaced by single spaces.

4) The Porter Stemmer [15] was used, with the following modification: an ignore list of more than 1000 stop words is defined and used to reduce the number of terms in the dataset.

5) Then, any remaining word consisting of just one character is removed.

The distribution of documents and terms, per class, for Reuters (R8) is shown in **Table 1**.

Table 1. Distribution of documents and terms among R8 classes.

R8					
Class #	Class	# train docs	# test docs	Total # docs	# terms
1	Acq	1596	696	2292	7323
2	Crude	253	121	374	2751
3	Earn	2840	1083	3923	7188
4	Grain	41	10	51	1038
5	Interest	190	81	271	1448
6	money-fx	206	87	293	1992
7	Ship	108	36	144	1676
8	Trade	251	75	326	2652
	Total	5485	2189	7674	13891 (after removing duplicates among classes)

4. Term Selection

It was proved in several researches that, in most of the cases, only a small, but proper, subset of the corpus terms is useful in classification [1,16,17]. Consequently, term selection is commonly used for reducing the dimensionality of a vector space in learning tasks. Some popular methods for measuring term strength in TC are information gain (IG), Chi-Square (χ^2), Document Frequency (DF), Odds Ratio (OR), Log Probability Ratio, Mutual Information (MI), and Term Strength (TS) [16-19].

4.1. Term Selection Method

Chi Square (χ^2) was used to compute the strength of each term in the corpus in this research. Chi square has shown to yield good results in classification, compared to other term selection methods [16-19].

The chi square score measures the correlation dependency between the term and its containing class. The higher this score is, the more discriminating the term is for that class. The chi square measure is computed for each term t in each class c_i as follows [20]:

$$\chi^2(t,c_i)=\frac{N\times(AD-CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \quad (8)$$

where:

N is the total number of training documents in the dataset,

A is the number of documents belonging to class c_i and containing t,

B is the number of documents belonging to class c_i but not containing t,

C is the number of documents not belonging to class c_i but containing t, and

D is the number of documents neither belonging to class c_i nor containing t.

The chi square measure can be globalized for terms that appear in more than one class (usually with different chi square measures in different classes) in one score by choosing the maximum or the average score. Finally, the reduced term set is chosen from the topmost chi square measure terms, ignoring terms with zero or small measures. Different term reduction criteria can be used, as explained in the next section.

4.2. Term Reduction Methods

Three different methods were used to reduce the result-

ing set of terms. Each of these methods was tried once keeping the infrequent terms in each document, and another time discarding them. This aims mainly to investigate the role of infrequent terms in enhancing document classification performance. Detailed results of using these reduced term sets in classification, as well as an analysis of these results, will follow in the subsequent sections.

4.2.1. Selecting the Topmost Terms from the Corpus as a Whole

Firstly, the topmost 100 chi square measure terms from the corpus as a whole (0.72% of the corpus terms) were selected. Another reduced term set is formed by selecting the topmost 70 chi square measure terms from the corpus as a whole (0.5% of the corpus terms). This term set was created to compare the classification performance using the same reduction method (the corpus topmost terms) but with a smaller number of terms.

4.2.2. Selecting an Equal Number of Terms from Each Class in the Corpus

An equal number of terms is chosen from each class as a second term reduction strategy. This aims to overcome the problem of the variation in the number of terms chosen from each class to build the classifier. The topmost 13 chi square measure terms were selected from each class, and these 104 terms were reduced to 96 terms (0.7% of the corpus terms) after elimination of duplicates.

4.2.3. Selecting an Equal Percentage of the Topmost Chi Square Measure Terms from Each Class

The last term reduction strategy experimented was to select an equal percentage of the topmost chi square measure terms from each class (0.5% of each class). The term set had 131 terms and these 131 terms were reduced to 108 terms after elimination of duplicates.

5. Term Weighting

In TC, each document is represented by a vector of term weights, which represent the strength or significance of these terms in this document. These weights are usually numbers which fall in the [0,1] interval. Several term weighting schemes were used in the literature of TC, such as document frequency (DF), Term Frequency (TF), Normalized Term Frequency.Inverse Document Frequency (tf.idf), Binary Weights, Information Gain (IG), and Weighted Inverse Document Frequency (WIDF). Three different term weighting schemes were experimented in this research: Chi square (χ^2), Normalized Term Frequency and Binary Weights. Chi Square measure was used as both a term selection criterion and a term weighting scheme. The chi square scores computed in the term selection stage were normalized using Min-Max normalization method, so as to map these scores to numbers in the range [0,1] by computing

$$s' = \frac{s - Min}{Max - Min} \tag{9}$$

where *Min* and *Max* denote the minimum and maximum chi square values respectively among all terms scores.

6. Performance Evaluation

The PN classifier performance was evaluated by computing its accuracy. Accuracy of a class c_i, Acc_i is computed as follows:

$$Acc_i = \frac{TP_i}{TP_i + FN_i + FP_i} \tag{10}$$

where

TP_i: True Positives with respect to a category c_i; the number of documents correctly claimed by the classifier as belonging to category c_i.

FP_i: False Positives with respect to c_i; the number of documents incorrectly claimed by the classifier as belonging to c_i.

FN_i: False Negatives with respect to c_i; the number of documents incorrectly claimed by the classifier as not belonging to c_i.

7. Results

The results reached for each reduced term set, using each weighting scheme, once keeping rare terms and another time discarding them are summarized in **Tables 2** and **3** respectively.

8. Analysis of Results

It was apparently clear from all the experiments in this research that keeping infrequent terms recorded a considerably much better performance compared with the results of the same term set and term weighting scheme with the rare terms being removed. This was valid for all the term sets regardless of the term reduction method used, or the term weighting scheme adopted. The enhancement on the accuracy recorded when keeping the rare terms is great; it reaches 17% in some experiments. **Figures 1-3** summarize the results related to this conclusion.

The Normalized Frequency term weighting scheme recorded the best performance among other weighting schemes on the four reduced term sets regardless of keeping or removing rare terms. Nevertheless, its performance when keeping rare terms recorded superior performance compared to the case when removing such terms. For the other two term weighting schemes tested, Binary Weights recorded better performance than Chi

Table 2. Accuracy keeping rare terms.

# Tokens	Term Weighting Scheme		
	Chi Square	Normalized Frequency	Binary
70	87.9397	92.4166	88.2595
96	84.2394	91.5943	82.6405
100	84.4221	91.7314	82.3207
108	81.6811	93.6044	78.2549

Table 3. Accuracy discarding rare terms.

# Tokens	Term Weighting Scheme		
	Chi Square	Normalized Frequency	Binary
70	74.4175	76.519	74.783
96	78.8945	80.9959	79.1686
100	77.7524	81.0873	78.3006
108	77.2042	82.9146	77.3413

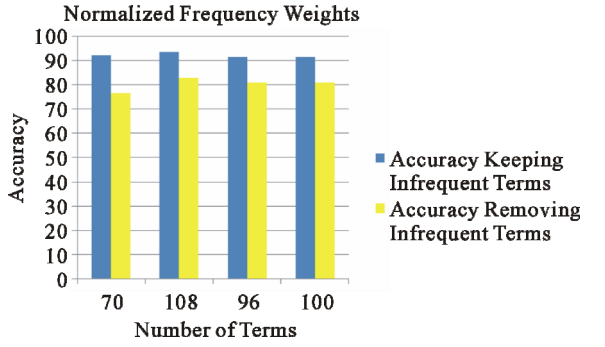

Figure 1. Keeping vs. removing rare terms using normalized frequency weights.

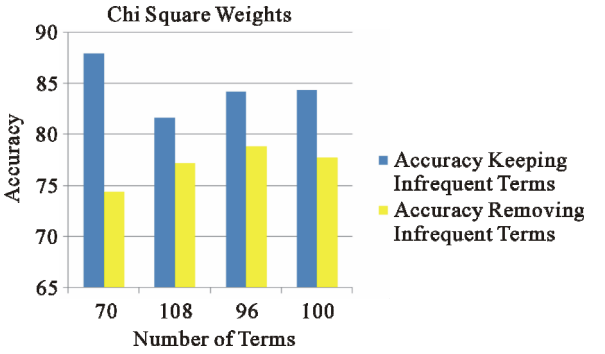

Figure 2. Keeping vs. removing rare terms using chi square weights.

Square when rare terms were removed, while Chi Square achieved better performance when such terms were considered. This can be attributed to the nature of the binary weighting scheme, which doesn't take into consideration the frequency of the term; it just records its presence or absence in the documents.

The 108-terms reduced term set has the optimum performance among the other term sets. This term set was constructed by selecting an equal percentage of the topmost terms in each class in the corpus. In fact, this complies with the conclusions in [1] which found that using equal percentage of class terms resulted in the best classification performance in several classifiers, compared with using equal number of terms from each class, or just choosing a specified number of the corpus topmost terms, as this guarantees that all classes are covered evenly in the term set selected for building classifiers. **Figures 4** and **5** summarize these results.

9. Conclusions

In this paper, the impact of keeping rare terms in enhancing Polynomial Networks (PN) based text categorization was investigated. Different term sets were used, which were selected using different reduction methods.

Furthermore, these sets were experimented with different term weighting schemes. All experiments were conducted once keeping the rare terms in the documents, and another time discarding them. Based on the results of the experiments conducted in this research, keeping infrequent terms is highly proposed in categorizing Reuters Data Set, due to their remarkable effect in enhancing the

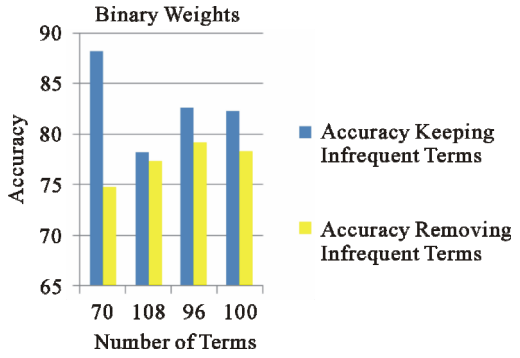

Figure 3. Keeping vs. removing rare terms using binary weights.

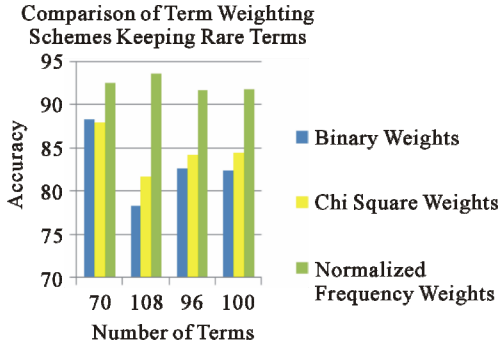

Figure 4. Comparison of term weighting schemes keeping rare terms.

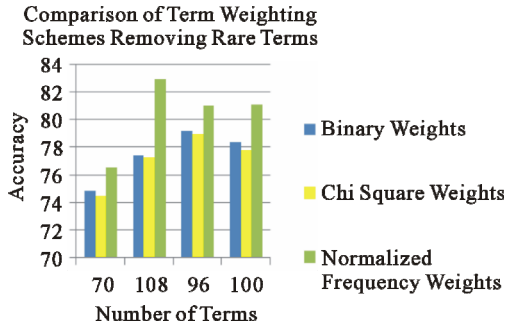

Figure 5. Comparison of term weighting schemes removing rare terms.

accuracy of automated text categorization, using different term weighting schemes on all the reduced term sets.

Furthermore, using normalized frequency as a term weighting scheme is proposed due to its superior performance and, at the same time, computation cost effectiveness. Finally, selecting the reduced term sets by choosing equal percentage of each class topmost terms is highly recommended, to eliminate the variation in the number of terms in classes, and to guarantee optimal text categorization performance.

The intended near future work is to extend the work conducted in this research to study the effect of keeping rare terms in text categorization using other well-known algorithms in the literature of text categorization, such as Support Vector Machines (SVM), Logistic Regression (LR), k-nearest-neighbor (kNN), Naive Bayes (NB), and the Radial Basis Function networks (RBF).

REFERENCES

[1] M. M. AL-Tahrawi and R. Abu Zitar, "Polynomial Networks versus Other Techniques in Text Categorization," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 22, No. 2, 2008, pp. 295-322. doi:10.1142/S0218001408006247

[2] R. Bekkerman, "Distributional Clustering of Words for Text Categorization," M.S. Thesis, Israel Institute of Technology, Haifa, 2003.

[3] D. Koller and M. Sahami, "Hierarchically Classifying Documents Using Very Few Words," *The* 14*th International Conference on Machine Learning* (*ICML*'97), Nashville, July 1997, pp. 170-178.

[4] D. Wang and H. Zhang, "Inverse-Category-Frequency based Supervised Term Weighting Scheme for Text Categorization," *Journal of Information Science and Engineering*, 2010.

[5] C. Deisy, M. Gowri, S. Baskar, S. M. A. Kalaiarasi and N. Ramraj, "A Novel Term Weighting Scheme MIDF for Text Categorization," *Journal of Engineering Science and Technology*, Vol. 5, No. 1, 2010, pp. 94-107.

[6] P. Schonhofen and A. A. Benczur, "Exploiting Extremely Rare Terms in Text Categorization," *Lecture Notes in Computer Science*, Vol. 4212, 2006, pp. 759-766.

[7] K. Fukunaga, "Introduction to Statistical Pattern Recognition," Academic Press, Cambridge, 1990.

[8] W. M. Campbell, K. T. Assaleh and C. C. Broun, "A Novel Algorithm for Training Polynomial Networks," *International NAISO Symposium on Information Science Innovations ISI*'2001, Dubai, March 2001.

[9] K. T. Assaleh and M. AL Rousan, "A New Method for Arabic Sign Language Recognition," Personal Communications, 2004.

[10] W. M. Campbell and C. C. Boun, "Using Polynomial Networks for Speech Recognition," Personal Communications, 2004.

[11] W. M. Campbell and K. T. Assaleh, "Polynomial Classifier Techniques for Speaker verification," *Proceedings of the International Conference on Acoustics*, *Speech*, and *Signal Processing*, Phoenix, 15-19 March 1999, pp. 321-324.

[12] K. T. Assaleh and W. M. Campbell, "Speaker Identification Using a Polynomial-Based Classifier," *International Symposium on Signal Processing and Its Applications*, Brisbane, 22-25 August 1999, pp. 115-118.

[13] G. H. Golub and C. F. Van Loan, "Matrix Computations," John Hopkins, Washington DC, 1989.

[14] Ana Site for Data Sets Suitable for Single-Label Text Categorization. http://www.gia.ist.utl.pt/~acardoso/datasets/

[15] M. F. Porter, "An Algorithm for Suffix Stripping," *Program*, Vol. 14, No. 3, 1980, pp. 130-137. doi:10.1108/eb046814

[16] G. Forman, "An Extensive Empirical Study of Term Selection Metrics for Text Classification," *Journal of Machine Learning Research*, Vol. 3, 2003, pp. 1289-1305.

[17] Y. Yang and J. Pederson, "A Comparative Study on Term Selection in Text Categorization," *Proceedings of the* 14*th International Conference on Machine Learning*, 1997, pp. 412-420.

[18] K. Fuka and R. Hanka, "Feature Set Reduction for Document Classification Problems," *IJCAI*-01 *Workshop*: *Text Learning*: *Beyond Supervision*, Seattle, August 2001, 2001.

[19] M. Rogati and Y. Yang, "High-Performing Feature Selection for Text Classification," *CIKM*'02, November 2002, pp. 4-9.

[20] Z. Zheng, X. Wu and R. Srihari, "Term Selection for Text Categorization on Imbalanced Data," *SIGKDD Explorations*, Vol. 6, No. 1, 2004, pp. 80-89. doi:10.1145/1007730.1007741

