A Review of Techniques to Determine the Optimal Word Score in Text Classification

Deepak Agnihotri, Kesari Verma, Priyanka Tripathi and Nilam Choudhary

Abstract Massive digital information is available in the form of text on Web pages and there is a continuous growth in this Web corpus. The resultant is a huge corpus with large number of dimensions in the form of text contents. Therefore, the classification of text corpus by its text contents is a challenging problem. Various feature selection techniques were used by the researchers to reduce the dimension of text corpus without affecting the performance of text classification. This paper investigates the importance of N-grams-based term indexing over unigram term indexing approach of text classification. It follows a new approach to find out the most informative words as features. Initially, a correlation score of each term for a class label is computed using the Pearson's correlation coefficient and then this score is multiplied with bigram collocation terms score which is computed by the chi-square method. The topmost n informative words are selected by sorting the words in descending order, where n is an empirically determined number. The performance of this approach is evaluated on two standard movie reviews text datasets using Naive Bayes Classifier. From the results, it is confirmed that the accuracy achieved by the proposed method is much better than state of the art.

Keywords Text mining · Feature selection · Text classification

D. Agnihotri · K. Verma
Department of Computer Applications, National Institute of Technology Raipur,
Raipur 492001, India
e-mail: dagnihotri.phd2012.mca@nitrr.ac.in

K. Verma
e-mail: kverma.mca@nitrr.ac.in

P. Tripathi (✉)
Department of Computer Engineering and Applications, National Institute
of Technical Teachers Training and Research, Bhopal, MP, India
e-mail: ptripathi@nitttrbpl.ac.in

N. Choudhary
Department of Computer Science and Engineering, Vivekananda Institute
of Technology (East), Jaipur, India
e-mail: neelamvit@gmail.com

G. M. Perez et al. (eds.), *Ambient Communications and Computer Systems*,
Advances in Intelligent Systems and Computing 696,
https://doi.org/10.1007/978-981-10-7386-1_43

1 Introduction

Availability of massive digital information is due to heavy use of e-corpus to represent news articles, researches, reviews, company information, etc. The growing corpus necessitates efficient computing tools and techniques for its effective management [2–6]. These computing tools and techniques use text contents of the corpus for its management. The word (or term) is the smallest constituent of text contents but it plays a lead role in text classification. The text contents are represented as word vectors, such as let a set of all words $\boldsymbol{t} = [\boldsymbol{t_1}, \boldsymbol{t_2}, \ldots, \boldsymbol{t_n}]$. Let $\boldsymbol{t_i}$ is the ith word of set t and it is represented by an ordered pair, such as $\boldsymbol{t_i} = (t_i, f_{ij})$, where first element of this ordered pair is the ith word itself and second element is the frequency of ith word in the jth document. This is the basic representation of words named bag-of-words (BOW) model which is used by the researchers in text mining [1, 7, 8, 10, 14–20]. This model represents a text which can be a sentence or a document as the bag of its individual words. This representation does not consider grammar and the order of word occurrence. It keeps only the frequency of word in the documents or sentences. The problem with this approach is that, if the two documents d_j and d_{j+l} have same terms, e.g., t_i, and t_{i+s} no matter in which order they are, both documents are considered as similar documents, where l and s are natural numbers.

The N-gram language model [8, 10, 16] addresses this issue, which is a set of various combination of terms from a given text corpus. This model counts frequency of combination of terms which occurred together in the sentences of various documents. Thus, it maintains the order of word occurrence in the sentences or documents, e.g., let a sentence, “I do not like the story of the movie.” Using the BOW model, its sentiment may be misclassified as positive because it contains a term “like.” In such cases, a combination of two or more terms, e.g., “not like” or “do not like” (i.e., N-grams) [14] helps in correct sentiment classification of documents.

Both BOW and N-gram models represent words as vectors but a document corpus may consist of so many words. Thus, it generates a huge dimension to deal with in text classification. A dimensionality reduction technique such as feature selection methods selects only the most informative features and ignore the rest [14]. This paper combines BOW and N-gram model to select the topmost n informative words as features, where n is an empirically determined number [3, 4, 6]. The words are arranged in descending order based on their score before selection of topmost n informative words. Initially, the correlation score of each term for a class label is computed using Pearson’s correlation coefficient and then it is multiplied with bigram collocation term’s score which is computed using chi-square (χ^2) method. The performance of this approach is evaluated on two standard movie reviews text datasets using Naive Bayes Classifier. From the results, it is confirmed that the accuracy achieved by the proposed method is much better than state of the art.

The rest of the paper comprises of six sections as follows, Sect. 2 deals with the works of other researchers in the same field and defines the research problem. Section 3 describes the research methodologies used in this paper. Description of the experimental setup is given in Sect. 4. Section 5 presents the results of the experiments with brief discussions. The paper concludes in Sect. 6 with some suggested future works.

2 Related Works

Substantial feature selection methods were discussed by the researchers in the area of text classification [15, 17–20]. Garcia Adeva et al. [1] used term frequency–inverse document frequency (TF–IDF) method to compute the score of terms systematic classification of reviews in medicine. Nanculef et al. [9] also used TF–IDF for comparison of the results. There are few works [8, 10, 16] related to N-gram term indexing used in text classification. The most used classifier in text classification is Naive Bayes.

2.1 Research Problem

Let a set of documents, $D = \{d1, d2, \ldots, dj|j > 0\}$, a set of classes of documents, $C_k = \{c_1, c_2, \ldots, c_k|k > 0\}$, and a set of terms as vectors $\boldsymbol{t} = [\boldsymbol{t_1}, \boldsymbol{t_2}, \ldots, \boldsymbol{t_n}]$. Given, $[D_{train}, C_k]$, where D_{train} is the training set corpus, and C_k is the k classes of documents. Now, the most common problem is the classification of test documents D_{test} such as $[D_{test}, C_k = ?]$. Since, in a conventional text corpus there are millions of terms and the representation of terms using BOW and N-grams model generates a large dimension. As a result, the Naive Bayes Text Classifier deals with a huge dimension. Few terms of this large dimension are required to discriminate the class label of the documents and many others disturbs the performance of classifier. Thus, the objective is to utilize the best parts of BOW and N-grams model and select the topmost n informative words which is passed as vocabulary to the Naive Bayes Classifier.

3 Methodology

The research methodology used in this paper is as follows: define the hypothesis, select the dataset, preprocessing of dataset, feature extraction, feature selection, classification, performance measure of the applied methods.

3.1 Hypothesis

Consider a document whose class is given by C. In movie review dataset, there are two classes, i.e., $C_k = \{pos, neg | k = 1, 0\}$. Let the hypothesis of this paper is that "N-gram-based term indexing approach along with χ^2 feature selection, and L-2 norm increases the accuracy of Naive Bayes Classifier in comparison of conventional BOW model for automatic text classification." The hypothesis is evaluated through experimental study.

3.2 DataSet

For experimental analysis, two movie reviews datasets [11, 12] are selected. Dataset1 [11–13] is movie review dataset and Dataset2 [12] is polarity dataset. Both datasets contain documents of reviews having positive and negative sentiments. Movie review dataset has 2000 text document files. Out of these 2000 text documents, 1000 documents have positive reviews and other 1000 have negative reviews. Polarity dataset includes sentence polarity dataset having 700 positive and 700 negative reviews sentiments.

3.3 Preprocessing

The raw text data is a sequence of tokens, e.g., words, numbers, spaces, punctuation marks, symbols, links, white spaces. This raw data needs to be preprocessed before classification, as most of the classifiers expect numerical feature vectors. The preprocessing steps are as follows: 1. Generation of tokens from text contents; 2. Feature extraction, i.e., removal of stop words, punctuation marks, numbers, and white spaces; 3. Counting the occurrences of tokens in each document; 4. Normalization of frequencies; 5. Vectorization of words, i.e., BOW or N-grams representation of words [2–4, 6].

3.4 Feature Selection

Feature selection is used to select the most informative words as features. In this context, initially, the score of each word is computed which is based on its frequency in the documents. The document frequency of the words is computed within each class which helps in computation of final score of the words. The words are arranged in descending order based on their final score, and the topmost n informative words are selected as features. Now each test document is classified based on the presence of these most informative words [2–4, 6]. The standard chi-square χ^2 method is the most proffered method for scoring of terms in text classification.

Mathematically, Pearson's correlation coefficient and chi-square testing both determines the correlation between word t_i and class C_j. If χ^2 $(t_i, C_j) = 0$, word t_i and class C_j are not correlated and t_i does not contain information to represent class C_j. Otherwise, the greater the value of the χ^2 (t_i, C_j) is, the more class information the word t_i owns. The mathematical equations for Pearson's correlation coefficient and chi-square testing are defined in Eqs. (1) and (2).

In the proposed method, initially, square of the Pearson's correlation coefficient is computed for each term associated with a class label using Eq. (2), then we multiply it with bigram collocation terms, referred as χ^2 score of the term for each class label in Eq. (3). Based on some threshold value, we find the optimal word score for a class label. The best class label for each word is computed by Eqs. (1) and (2).

$$\rho^2(t_i, C_j) = \frac{(a_{ij} \times d_{ij} - b_{ij} \times c_{ij})^2}{(a_{ij} + b_{ij}) \times (a_{ij} + c_{ij}) \times (b_{ij} + d_{ij}) \times (c_{ij} + d_{ij})} \tag{1}$$

$$\chi^2(t_i, C_j) = \frac{N \times (a_{ij} \times d_{ij} - b_{ij} \times c_{ij})^2}{(a_{ij} + b_{ij}) \times (a_{ij} + c_{ij}) \times (b_{ij} + d_{ij}) \times (c_{ij} + d_{ij})} \tag{2}$$

where N is the total number of documents; a_{ij} is the frequency that feature t_i and class C_j co-occur; b_{ij} is the frequency that feature t_i occurs and does not belong to class C_j; c_{ij} is the frequency that class C_j occurs and does not contain feature ti; d_{ij} is the number of times when neither C_j nor t_i occurs.

Unit Vector ($\hat{v}$) of the vector ($\boldsymbol{v}$) is computed to normalize a vector, such as $\hat{v} = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_p}$. The unit vector is a vector of length 1. Let the $\|\boldsymbol{v}\|_p$ is the norm (magnitude, length) of the vector $\boldsymbol{v}$ in the L^p space. Then, Lp-norm is as, $|\boldsymbol{u}\|_p = (|u_1|^p + |u_2|^p + |u_3|^p + \cdots + |u_n|^p)^{\frac{1}{p}}$ and it in simplified form as: $|\boldsymbol{u}\|_p = (\sum_{i=1}^{n} |\boldsymbol{u}_i|^p)^{\frac{1}{p}}$. An L2-norm, is the Euclidean norm, i.e., a norm with p = 2. It is the most common norm used to measure the length of a vector (i.e., magnitude). It is used, when we have an unqualified length measure (without the p number). Any norm can be used to normalize the vector, but L2-Norm [14] is the most common in the text mining.

It is very common in text classification to use the term frequency–inverse document frequency (*TF–IDF*) transformation in order to re-weight the count features into floating point values, which is suitable for use by a classifier. It is originally a term weighting scheme which scales up frequent terms and scales down rare terms. It also addresses the issues due to keyword spamming. Its mathematical expression is as follows:

$$W_{i,j} = tf_{i,j} * \log \frac{N}{df_i} \tag{3}$$

where $W_{i,j}$ = weight for term i in document j. N = total number of documents in the corpus, $tf_{i,j}$ = Term frequency of term i in document j, df_i = document frequency of term in the corpus. In the N-gram-based term indexing approach for words of length 1–2.

$$W_{(i,i+1),j} = tf_{(i,i+1),j} * \log \frac{N}{df_{(i,i+1)}} \tag{4}$$

where $W_{(i,i+1),j}$ = weight for term (i,i+1) in document j. N = total number of documents in the corpus, $tf_{(i,i+1),j}$ = term frequency of term (i,i+1) in document j, $df_{(i,i+1)}$ = document frequency of term (i,i+1) in the corpus.

3.5 Performance Evaluation

Given, $< D, C >$, where D is document set and C is a set of all classes of documents. In supervised classification, the text corpus is divided into training set and test set. During training phase, each document is assigned a class through this information machine is trained. After training, test set is passed to the machine, the class of the documents is not known to the machine at this time, based on training corpus machine assigns a class to each document. The classification accuracy can be measured by comparing the assigned classes with actual classes of each document. Precision, recall, and F1-measure of the classifier are measured as follows:

$$Precision = \frac{tp}{tp + fp} \tag{5}$$

$$Recall = \frac{tp}{tp + fn} \tag{6}$$

$$F_1 measure = 2 \times \frac{Precision * Recall}{Precision + Recall} \tag{7}$$

4 Experimental Setup

For experimentation, initially, 1000 positive and 1000 negative documents are combined as a corpus with 2000 documents. Further, it is divided into two parts, first part contains 1500 documents as training corpus and other 500 documents as test corpus. Similar steps are followed with polarity dataset which consists of 700 positive and 700 negative sentiments.

Training Phase:
Step 1: Determine the vocabulary, i.e., the N-grams of length (1, 2) from corpus using Eq. (5). Let $|Vocabulary| = |V|$= total number of N-grams of length (1, 2).
Step 2: For each *i*th word w_i in the vocabulary V, compute the probability $P(w_i|C_k)$ of the word w_i occurring with class C_k. Steps for computation of this value:-

START

1. Combine the documents as per class label into one text file.
2. Count how many words occurred in the file, call it n.
3. For each word w_i in the vocabulary V, count how many times it occurred in the text file and call it n_i.
4. To obtain word score for each word w_i in the vocabulary V create a method using Eqs. (2) and (3). The TF–IDF weight can be obtained using Eqs. (4) and (5).
5. Finally, apply Naive Bayes Classifier for each word w_i occurring with class C_k as:

$$P(w_i|C_k) = \frac{n_i + 1}{n + |v|} \tag{8}$$

Test Phase:

Step 3: Create a method to find performance measures using Eqs. (6), (7), and (8).
Step 4: plot the graph to analyze the result.
END

5 Results and Discussions

All the experiments have been conducted on UBUNTU 14.04 LTS 32-bit environment using Python 2.7.6 Language. The details of the experimental results are as follows, the multinomial Naive Bayes Classifier has been applied on two datasets Dataset1 and Dataset2 by using bag-of-words (BOW), N-gram with χ^2 feature selection, and our proposed method. The performance measures have been obtained in the form of precision, recall, F1-measure, and accuracy, shown in Tables 1, 2, 4, and 5. Table 3 shows the comparison of results and Fig. 1a, b presents a graphical visualization of compared result.

Table 1 Without N-gram (single word features)

Dataset (Class)	Precision	Recall	F1 measure
Dataset1 (Pos)	0.652	0.98	0.783
Dataset1 (Neg)	0.960	0.476	0.636
Dataset2 (Pos)	0.585	0.989	0.735
Dataset2 (Neg)	0.963	0.297	0.454

Table 2 With N-gram using χ^2 feature selection (dataset1)

Class	Precision	Recall	F1-score	Support
Neg	0.84	0.75	0.80	244
Pos	0.77	0.85	0.81	256
Avg/total	0.81	0.80	0.80	500

Table 3 Comparison of Naive Bayes classifiers accuracy

Model	Max. accuracy (changing features count)	
	Dataset1	Dataset2
BOW (%)	73–81	64–73
N-gram (%)	81–85	81–83
Proposed method (%)	87–93	84–92

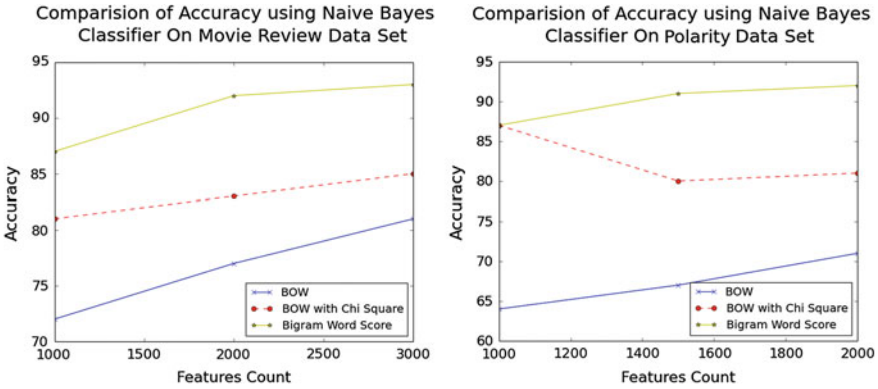
Table 4 Performance measures of proposed method in dataset1 (movie review data)

Performance measures	Best words count (when best bigrams = 200)					
	1000	2000	4000	6000	8000	10000
Pos (precision)	0.914	0.929	0.922	0.917	0.903	0.913
Pos (recall)	0.808	0.848	0.896	0.924	0.932	0.928
Neg (precision)	0.828	0.860	0.899	0.923	0.930	0.927
Neg (recall)	0.924	0.936	0.924	0.916	0.9	0.912
Pos (F1 measure)	0.858	0.887	0.909	0.920	0.917	0.921
Neg (F1 measure)	0.873	0.897	0.911	0.920	0.915	0.919
Avg/total accuracy (%)	86.6	89.2	91	92	91.6	92

Table 5 Performance measures of proposed method in dataset2 (polarity data)

Performance measures	Best words count (when best bigrams = 200)					
	1000	2000	4000	6000	8000	10000
Pos (precision)	0.851	0.875	0.879	0.883	0.893	0.888
Pos (recall)	0.817	0.88	0.914	0.949	0.954	0.954
Neg (precision)	0.824	0.879	0.911	0.944	0.951	0.951
Neg (recall)	0.857	0.874	0.874	0.874	0.886	0.88
Pos (F1 measure)	0.834	0.878	0.896	0.915	0.923	0.920
Neg (F1 measure)	0.840	0.877	0.892	0.908	0.917	0.914
Avg/total accuracy (%)	83.71	87.71	89.43	91.14	92	91.71

Table 1 is used to show the performance measures of Naive Bayes Classifier on Dataset1 and Dataset2 by taking single word as features and without applying any feature selection techniques. An average 72.8% accuracy has been observed for Dataset1 and 64.29% for Dataset2. Further, N-grams of length 1–2 have been selected by χ^2 feature selection method which uses threshold values as 1000, 1500, 2000, etc. Table 2 shows performance of the features obtained by this step on Dataset1. The Naive Bayes Classifier has given an average 81% accuracy. Tables 4 and 5 show the performance measures obtained by applying the proposed method with Naive Bayes Classifier on Dataset1 and Dataset2. It has given 87–93% accuracy on Dataset1 and 84–92% accuracy on Dataset2. The comparison of the results



(a) Movie Review Data.

(b) Polarity Data.

Fig. 1 Comparison of performance

is shown by Table 3, in which 73–81% accuracy has been achieved by BOW model, 81–85% accuracy by N-grams and χ^2 feature selection method, and 87–93% accuracy by the proposed method in Dataset1.

Similar results have been obtained on Dataset2, which has given 64–73% accuracy by BOW model, 81–83% accuracy by N-grams and χ^2 method, and 84–92% accuracy by the proposed method. The visualization of the comparison results for both datasets is shown in Fig. 1a, b, where the blue color line is for single word feature (BOW), while the red color line is for N-gram model with χ^2 feature selection, and on the top with yellow color is the proposed method. As it can be observed from these figures that proposed method is more accurate than other two approaches. Thus, the hypothesis of this paper, i.e., "N-gram based term indexing approach along with χ^2 feature selection, and L-2 norm increases the accuracy of Naive Bayes Classifier in comparison of conventional BOW model for automatic text classification" is true.

6 Conclusions and Future Works

This paper investigated the importance of N-grams-based term indexing over unigram term indexing approach of text classification. It followed a new approach to find out the most informative words as features. Initially, a correlation score of each term for a class label has been computed using the Pearson's correlation coefficient, and then this score is multiplied with bigram collocation terms score which has been computed by the chi-square method. The topmost n informative words have been selected by sorting the words in descending order, where n is an empirically determined number. We created hypothesis, i.e., "N-gram-based term indexing approach along with χ^2 feature selection, and L-2 norm increases the accuracy of Naive Bayes Classifier in comparison of conventional BOW model." We have applied Naive Bayes Classifier on two review datasets containing positive and negative sentiments as two class labels. We have applied various popular methods like (1) single words as features (BOW), (2) N-gram along with χ^2 feature selection (3), and our proposed method to find the optimal word score for each class label. From the experimental results, the validity of our hypothesis is assured. We get better performance measures in terms of precision, recall, F1-measure, and accuracy for both datasets in comparison of other methods. It has been tested with datasets having two class labels only, this might be the limitation of the proposed method. In future, it can be checked with multiclass documents. Other classifiers, viz. KNN, SVM, Rochhio, and Random Forests, can be used with this approach.

References

1. Adeva, J.G., Atxa, J.P., Carrillo, M.U., Zengotitabengoa, E.A.: Automatic text classification to support systematic reviews in medicine. Expert Systems with Applications 41(4), 1498–1508 (2014)
2. Agnihotri, D., Verma, K., Tripathi, P.: Pattern and cluster mining on text data. In: In Fourth International Conference on Communication Systems and Network Technologies. pp. 428–432. IEEE Computer Society, CSNT, Bhopal (2014)
3. Agnihotri, D., Verma, K., Tripathi, P.: Computing correlative association of terms for automatic classification of text documents. In: Proceedings of the Third International Symposium on Computer Vision and the Internet. pp. 71–80. ACM (2016)
4. Agnihotri, D., Verma, K., Tripathi, P.: Computing symmetrical strength of n-grams: a two pass filtering approach in automatic classification of text documents. SPRINGERPLUS 5(942), 1–29 (2016)
5. Agnihotri, D., Verma, K., Tripathi, P.: An empirical study of clustering algorithms to extract knowledge from pubmed articles. Transactions on Machine Learning and Artificial Intelligence 5(3), 13 (2017)
6. Agnihotri, D., Verma, K., Tripathi, P.: Variable global feature selection scheme for automatic classification of text documents. Expert Systems with Applications, Elsevier 81, 268–281 (2017), http://www.sciencedirect.com/science/article/pii/S0957417417302208
7. Azam, N., Yao, J.: Comparison of term frequency and document frequency based feature selection metrics in text categorization. Expert Systems with Applications 39(5), 4760–4768 (2012)
8. Geiger, W.M., Rauch, J., Mair, P., Hornik, K.: Text Categorization in R: A Reduced N-Gram Approach, pp. 341–349. Springer Berlin Heidelberg, Berlin, Heidelberg (2012), https://doi.org/10.1007/978-3-642-24466-7_35
9. Nanculef, R., Flaounas, I., Cristianini, N.: Efficient classification of multi-labeled text streams by clashing. Expert Systems with Applications 41(11), 5431–5450 (2014)
10. Nbviewer: Document-level text analysis (2013), http://nbviewer.ipython.org/gist/rjweiss/7158866
11. NLTK-Project: nltk movie review corpus (2014), http://www.nltk.org/nltk_data/
12. Pang, B., Lee, L.: Movie review data (2005), http://www.cs.cornell.edu/People/pabo/movie-review-data/
13. Perkins, J.: Text classification for sentiment analysis using naive bayes classifier (2010), http://streamhacker.com/2010/05/10/
14. Perone, C.S.: Machine learning text feature extraction (tf-idf) (2011), http://pyevolve.sourceforge.net/wordpress/?p=1747
15. Pinheiro, R.H., Cavalcanti, G.D., Correa, R.F., Ren, T.I.: A global-ranking local feature selection method for text categorization. Expert Systems with Applications 39(17), 12851–12857 (2012)

16. Rahmoun, A.: Experimenting n-grams in text categorization. International Arab Journal of Information Technology 4, 377–385 (October 2007), http://iajit.org/PDF/vol.4,no.4/13-Rahmoun.pdf
17. Ren, F., Sohrab, M.G.: Class-indexing-based term weighting for automatic text classification. Information Sciences 236, 109–125 (2013)
18. Shang, C., Li, M., Feng, S., Jiang, Q., Fan, J.: Feature selection via maximizing global information gain for text classification. Knowledge-Based Systems 54, 298–309 (2013)
19. Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., Wang, Z.: A novel feature selection algorithm for text categorization. Expert Systems with Applications 33, 1–5 (2007)
20. Yao, Z., Zhi-Min, C.: An optimized nbc approach in text classification. Physics Procedia 24, 1910–1914 (2012)