Feature Extraction for Co-Occurrence-Based Cosine Similarity Score of Text Documents

Ammar Ismael Kadhim[1,2], IEEE, Yu-N Cheah[1], Member, IEEE and Nurul Hashimah Ahamed[1], Member, IEEE
[1]School of Computer Sciences
Universiti Sains Malaysia
Penang, Malaysia
[2]Department of Computer Science
College of Medicine
Baghdad, Iraq
ammarusm70@gmail.com, yncheah@cs.usm.my, and nurulhashimah@cs.usm.my

Lubab A. Salman
College of Engineering
Alnahrain University
Baghdad, Iraq

***Abstract*— A major challenge in topic classification (TC) is the high dimensionality of the feature space. Therefore, feature extraction (FE) plays a vital role in topic classification in particular and text mining in general. FE based on cosine similarity score is commonly used to reduce the dimensionality of datasets with tens or hundreds of thousands of features, which can be impossible to process further. In this study, TF-IDF term weighting is used to extract features. Selecting relevant features and determining how to encode them for a learning machine method have a vast impact on the learning machine methods ability to extract a good model. Two different weighting methods (TF-IDF and TF-IDF Global) were used and tested on the Reuters-21578 text categorization test collection. The obtained results emerged a good candidate for enhancing the performance of English topics FE. Simulation results the Reuters-21578 text categorization show the superiority of the proposed algorithm.**

Keywords—feature extraction; topic classification; cosine similarity score; TF-IDF weighting.

I. INTRODUCTION

Feature extraction (FE) is one of the dimensional reduction that are commonly used on datasets with tens or hundreds thousand of features [1]. Dimension reduction is a one of the process is used in different applications such as data mining and retrieval information. The main goal of dimension reduction is to reduce high-dimensional data in a lower-dimensional subspace, while essential features of the original data are kept as much as possible [2]. The biggest challenge in feature extraction comes from numerous variations of key terms (feature) for each document in dataset [3]. They are the text preprocessing steps, the correlation indexing methods as well as the similarity measure [4].

This paper first addresses the problem of extraction of key words or key terms from the document content, i.e. the Reuters-21578 text categorization test collection. These key terms (features) are subsequently analyzed and term relations are detected. Four methods for generating key words were employed: text preprocessing, indexing techniques feature extraction, and dimension reduction.

The use of cosine similarity score in dimension reduction leads to an effective learning algorithm which can enhance the generalization ability of feature extraction.

II. RELATED WORK

Different methods to measure the cosine similarity score between two documents in a dataset have been evaluated in previous studies [4]. Specifically, many studies have focused on a comparison between probabilistic methods and the existing text cosine similarity measures in a document. For example, Forman introduced an extensive comparative study of feature selection metrics for the high-dimensional domain of topic classification (TC), focusing on support vector machines and 2-class problems, typically with high class skew. It revealed the surprising performance of a new feature selection metric, Bi-Normal Separation. It also found a novel evaluation methodology that considers the common problem of trying to choose one or two metrics that have the best chances of getting the best performance for a given dataset. Somewhat surprisingly, choosing the two perfect performing metrics can be sub-optimal: when the best metric fails, the other may have associated failures, as it is the case for information gain (IG) and Chi square for maximizing precision [5].

There are two kinds approaches to learn term such as supervised and unsupervised, Debole and Sebastiani suggested supervised term weighting (STW), a term weighting methodology specifically designed for IR applications relating supervised learning, such as TC and text filtering. Related to this, supervised term indexing leverages on the training data by weighting a term according to how different its distribution is in the positive and negative training examples. In addition, they introduce that this should get the form of replacing inverse document frequency (IDF) by the category based term evaluation function that has previously been used in the term selection phase; as such, STW is also efficient, since it reuses

978-1-4799-6428-4/14/$31.00 ©2014 IEEE

for weighting purposes the scores already calculated for term selection purposes, while Soucy and Mineau who suggest a new technique (ConfWeight) to weight features in the vector space model for text classification by leveraging the classification process [6].

So far, the most commonly used method is TF-IDF, which is unsupervised [7]. However, there have been little discussions about TC learning, Ikonomakis et al. presented that automated TC has been considered as a vital method to manage and process a vast amount of documents in digital forms that are common and continuously increasing. In general, TC plays an essential role in information extraction and summarization, text retrieval, and question answering [8]. Some researchers have studied the relevant subject and the researcher will shed light on some of these relevant studies as follow: Kamruzzaman and Haider presented a new methodology for TC that requires fewer documents for training. The researchers used word relation, i.e. association rules from these words are used to derive feature set from pre-classified text documents [9].

Similar studies have been conducted by Shi et al. studied TC, that is an important sub-task in the field of text mining. By considering the frequency, dispersion and concentration concurrently, an enhanced feature selection, feature weighting and term reduction method tackling large set of Chinese texts in the real world was achieved [10].

III. PROPOSED APPROACH

The proposed approach can be divided into four main stages such as: preprocessing stage, indexing techniques stage, feature extraction and dimensional reduction as shown in Figure 1.

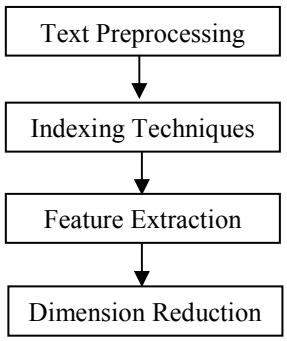


Figure 1. The stages of proposed system.

A. Text Preprocssing Stage

The first stage of the proposed approach is to convert the original textual data into a raw data structure, where the most significant text features that serve to distinguish between text categories are identified. This stage is the most critical and complex leading to the representation for each text document through the selection of a set of index terms. The major objective of text preprocessing is to obtain the key features or key terms from the text document dataset and to improve the relevancy between word and document, and the relevancy between word and class.

After reading the input text documents, the text preprocessing texts stage divides the text document to features called tokens, words, terms or attributes. These represent the text document in the form of a vector space which consists of the features and their weights. The weights are obtained by the frequency of each feature in that text document. Following this, non-informative features such as stop words, numbers and special characters are removed. Next, the remaining features are standardized by reducing them to their root using the stemming process. Despite the removal of the non-informative features and the stemming process, the dimensionality of the feature space may still be too high. Therefore, a threshold value is applied to reduce the size of the feature space for each input text document based on the frequency of each feature in that text document. These steps mentioned above are used to prepare the text document as depicted in Figure 2.

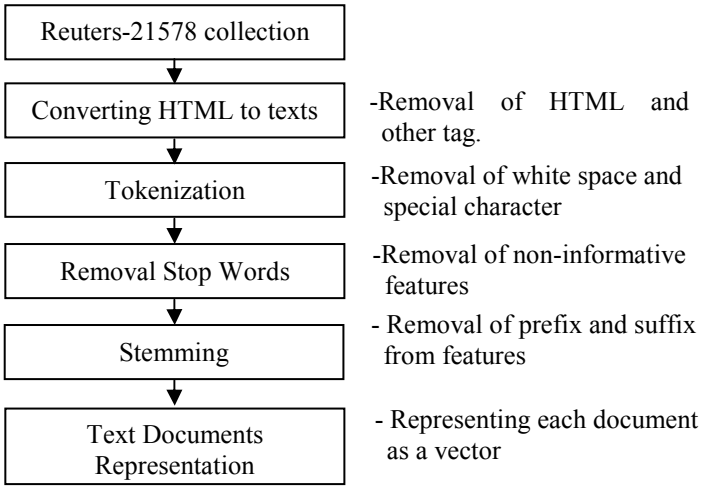


Figure 2. The stages of proposed system

B. Indexing Techniqes Stage

The major objective of document indexing is to increase the efficiency by extracting from the resulting document a selected set of terms to be used for indexing the document. Document indexing involves choosing the suitable set of keywords based on the whole corpus of documents, and assigning weights to those keywords for each particular document, thus transforming each document into a vector of keyword weights. The weight normally is related to the frequency of occurrence of the term in the document and the number of documents that use that term.

C. Feature Extraction

FE is an important approach it provides a lot of information regarding the text documents such as the highest and lowest term frequency for each document. Selecting relevant features and determining how to encode them for a learning machine method can have a vast impact on the learning machine methods ability to extract a good model. In this study, will use TF-IDF terms weighting is to extract features. Define an keywords set for each text document category, Transform Text to Numerical Feature by counting the number of occurrences (called term frequency) in the text document. This process is

vital in assigning the quality of the next stage that is the dimension reduction stage. It is important to choose the intended keywords that carry the meaning, and refuse the words that do not support to recognizing between the documents [11]. Extracting collection features by combining individual document features. These steps mentioned above are depicted in Figure 3.

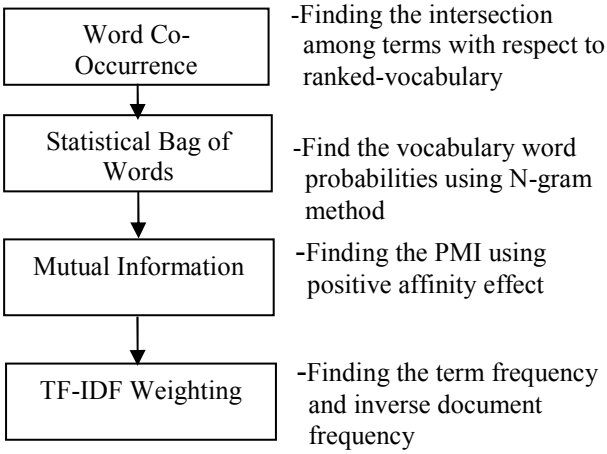


Figure 3. Indexing technique feature extraction stages.

Term weighting can be as simple as binary representation or as detailed as a mix of terms and existing datasets. TF-IDF is the most widely known and used weighting method, and it is remain even comparable with novel methods. The purpose of the text preprocessing stage is to refer to each document as a feature vector that is to divide the text into individual words. In TF-IDF term weighting, the text documents are characterized as transactions. Choosing the keyword for the feature selection process is the main preprocessing process necessary for the indexing of documents. This study utilized two different TF-IDF methods, i.e. Global and normal, to weight the terms in term-document matrices of our evaluation datasets. A common practice to avoid this variability or, at least, reduce the possible impacts resulting from it, is the normalization of the TF-IDF scores for each document in the collection by using the Euclidean norm is calculated by using Equations (1) and (2) respectively [12] as follows:

$$\text{TF-IDF}=\ln(\text{TF}) \times \ln(\text{IDF}) \quad (1)$$

$$\text{TF-IDF}=\text{TF} \times \text{IDF} \quad (2)$$

D. Dimension Reduction Stage

In this study, the researcher used the cosine similarity score function between a pair of vectors depends upon the word frequency and number of common words in both text documents. So the cosine similarity values of two vectors in the training dataset are found. The cosine similarity between a two vectors is computed in Equation (3) [4] as follows:

$$sim(a,b) = \frac{\vec{a} \bullet \vec{b}}{|\vec{a}| \times |\vec{b}|} = \frac{\sum_{=1}^{m} \sum_{j=1}^{N_i} a_{ij} \times b_{ij}}{\sqrt{\sum_{=1}^{m} \sum_{j=1}^{N_i} a^2{}_{ij}} \sqrt{\sum_{=1}^{m} \sum_{j=1}^{N_i} b^2{}_{ij}}} \quad (3)$$

Where $|\vec{a}|$ and $|\vec{b}|$ are the norms of the document vectors, a_i is the TF-IDF weight of document i, b_i is the TF-IDF weight of document $i+1$ in the dataset, m is the number of document in the features set, and N_i is the number of the terms belong to document i in the features set. Since $a_{ij} \geq 0$ and $b_{ij} \geq 0$, (a,b) varies from zero to +1 and shows the degree of similarity between two text documents for each class.

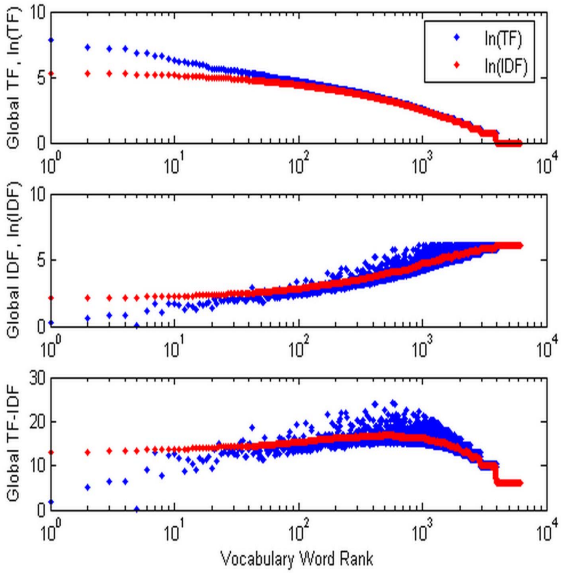
IV. Experimental Results

A. Dataset

The dataset chosen was the Reuters-21578 text categorization test collection consisting 21578 news documents. Only the 10 most popular classes are used from the 135 categories assigned to the documents, and only those text documents uniquely placed in one of these classes were considered.

B. Performance Evaluation

In this section, we present an experimental evaluation produced by the TF-IDF and TF-IDF Global techniques based on cosine score similarity function using on the Reuters-21578 text categorization test collection. Figure 4 shows the relationship between vocabulary word rank and the TF-IDF Global technique using logarithmic values, while Figure 5 shows the TF-IDF Global without using logarithmic values. Figure 6 shows the histogram of cosine score similarity after eliminating the high-frequency terms (or stop words), while Figure 7 shows the histogram using the same similarity function after applying all the preprocessing steps, which resulted in the reduction of features of 10-20% compared with Figure 6.



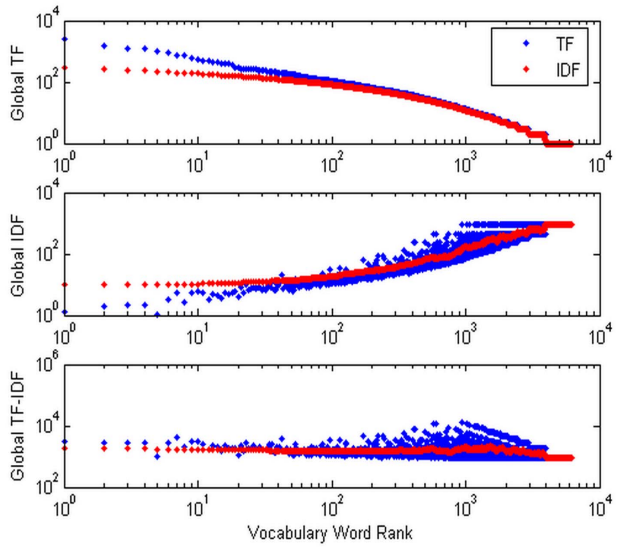


Figure 4. The relationship between vocabulary word rank and TF-IDF weighing using logarithm function.

Figure 5. The relationship between vocabulary word rank and TF-IDF Global weighting (non-logarithmic).

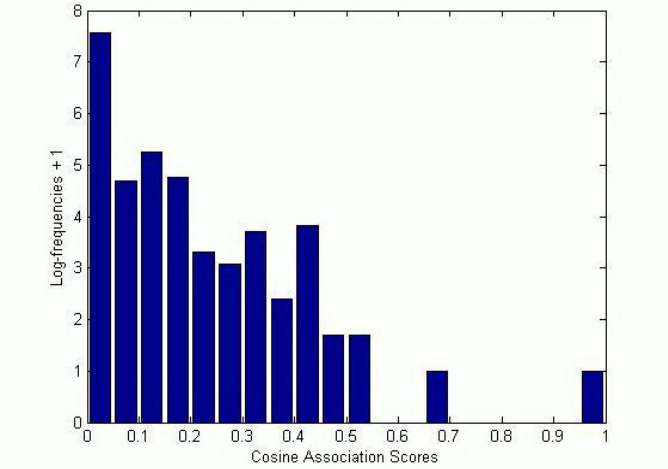


Figure 6. The histogram of cosine similarity score after eliminating the high-frequency terms (or stop words)

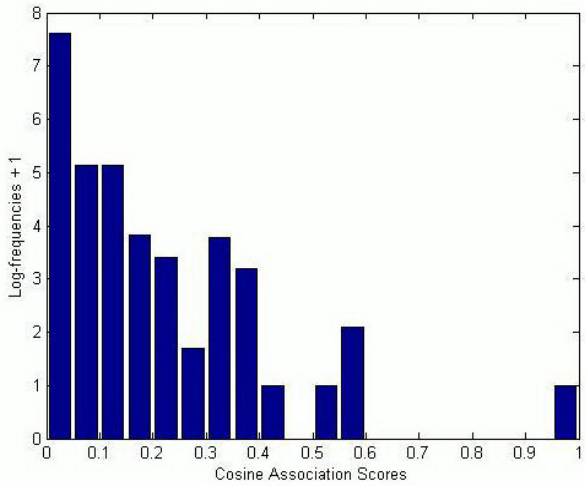


Figure 7. The histogram of cosine similarity score after applying all the preprocessing steps

V. CONCLUSION

The present work uses text preprocessing, indexing techniques, feature extraction and dimension reduction to extract the key terms (features) based on the Reuters-21578 text categorization test collection. From the results, it could be seen that preprocessing text stage and features weighting have a huge effect on performances of feature extraction. The reprocessing and features weighting are to reduce the number of features which was successfully met by the given techniques. Thus, the findings of the present study are cosine score similarity to improve and to reduce the high dimensionality of feature spaces. The results showed that the removal of stop-words can enhance the recognition degree between documents and the system performance for feature extraction.

References

[1] S. Ayache, G. Quénot, and J. Gensel, "Classifier fusion for SVM-based multimedia semantic indexing." Springer: Berlin-Heidelberg, 2007, pp. 494-504.

[2] AT. Sadiq, and SM. Abdullah. "Hybrid Intelligent Technique for Text Categorization.", IEEE, pp. 238-245, 2012.

[3] H. Benfriha, F. Barigou, and B. Atmani, "Lattice-cell: Hybrid approach for text categorization." arXiv preprint arXiv, 2013, pp. 1311-4151.

[4] D. Metzler, S. Dumais, and C. Meek. "Similarity measures for short segments of text." Springer Berlin Heidelberg, 2007, pp. 6-27.

[5] G. Forman, "An extensive empirical study of feature selection metrics for text classification." The Journal of machine learning vol. 3 2003 pp. 1289-1305.

[6] P. Soucy, GW and Mineau, "Beyond TFIDF weighting for text categorization in the vector space model." In (IJCAI) vol. 5, 2005, pp. 1130-1135.

[7] PD Turney and P Pantel. "From frequency to meaning: Vector space models of semantics." Journal of artificial intelligence research vol. 37.1, 2010, pp.141-188.

[8] M. Ikonomakis, S. Kotsiantis, and V. Tampakas. "Text classification using machine learning techniques." WSEAS Transactions on Computers vol. 4.8, 2005, pp. 966-974.

[9] SM. Kamruzzaman and F. Haider, "A hybrid learning algorithm for text classification." arXiv preprint arXiv, 2010, pp.1009-4574.

[10] K. Shi, et al. "Efficient text classification method based on improved term reduction and term weighting." The Journal of China Universities of Posts and Telecommunications vol. 18, 2011, pp. 131-135.

[11] F. Debole, and F. Sebastiani, "Supervised term weighting for automated text categorization." Text mining and its applications. Springer Berlin Heidelberg, 2004, pp. 81-97.

[12] S. Ramasundaram, and SP. Victor,"Text Categorization by Backpropagation Network." International Journal of Computer Applications, vol. 8(6), 2010, pp. 1-5.

.