Improvement of Sentiment Analysis Based on Clustering of Word2Vec Features

Eissa M.Alshari
Computer and Information Techonology
Ibb university, Yemen
eissa.alshari@student.upm.edu.my

Azreen Azman*, Shyamala Doraisamy,
Norwati Mustapha and Mustafa Alkeshr
Universiti Putra Malaysia, Serdang, Malaysia
{azreenazman,shyamala,norwati}@upm.edu.my,
mostafa.alksher@student.upm.edu.my

***Abstract*—Recently, many researchers have shown interest in using Word2Vec as the features for text classification tasks such as sentiment analysis. Its ability to model high quality distributional semantics among words has contributed to its success in many of the tasks. However, due to the high-dimensional nature of the Word2Vec features, it increases the complexity of the classifier. In this paper, a method to construct a feature set based on Word2Vec is proposed for sentiment analysis. The method is based on clustering of terms in the vocabulary based on a set of opinion words from a sentiment lexical dictionary. As a result, the feature set for the classification is constructed based on the set of clusters. The effectiveness of the proposed method is evaluated on the Internet Movie Review Dataset with two classifiers, namely the Support Vector Machine and the Logistic Regression. The result is promising, showing that the proposed method can be more effective than the baseline approaches.**

***Index Terms*—Sentiment analysis, Word2Vec, Word embeddings, Clustering**

I. Introduction

Recently, there is an explosive number of user reviews or comments on products and services available on the Web and social media [1]. It has become the source of information for users in making everyday decision, especially on choosing a product to buy or a movie to watch [2]. Due to the huge number of different opinions on a certain product or service, a user may find it difficult to summarize the overall sentiment based on those reviews or comments.

Over the years, researchers have developed different techniques for sentiment analysis to classify the reviews or comments into their polarity classes such as positive, negative or neutral [3], [4]. Several machine-learning techniques such as Logistic Regression (LR), Support Vector Machine (SVM) and Naïve Bayes have shown to be effective in this text classification problem [5]. The effectiveness of such techniques relies on the features used in the classification task. Several features have been investigated for this task such as the bag-of-words (BoW), lexical and syntactic features [6].

Since the introduction of Word2Vec by Mikolov $et\ al.$ [7], [8], [9] to discover semantic relation between words, it has been used as features for several text classification tasks [10]. Due to the high-dimensional nature of the Word2Vec features, it increases the complexity of the classifier. Several feature extraction methods can be applied in order to reduce the dimension of the Word2Vec features [11].

In this paper, a method to construct feature set is proposed to reduce the dimension of the Word2Vec features for sentiment analysis. In particular, the set of terms in a vocabulary are clustered around opinion words in order to distribute them based on polarity. It is believed that such a method will improve the effectiveness of sentiment classification of text.

This paper is organized as follows. A review of related work on sentiment analysis and word embedding is presented in Section II. The proposed feature extraction method for Word2Vec based on clustering is explained in Section III. In Section IV, the experimental results are analyzed and elaborated. Finally, the conclusion and future work are discussed in Section V.

II. Related Work

Sentiment analysis (SA) is a collection of methods to determine the polarity or orientation (positive, negative or neutral) of sequence of words in a text [12]. Many techniques and type of features have been investigated for SA including the use of bag-of-words (BoW) model as the feature for the classification [13]. The BoW is an approach to model texts numerically in many text mining and information retrieval tasks [14]. Several weighting schemes have been successfully used in the BoW such as the n-gram, Boolean, co-occurrence, tf and $tf.idf$ [4], [15].

In the context of modeling distributional semantics within text, several models were proposed for estimating continuous representations of words, such as the Latent Semantic Analysis (LSA) [16], the Latent Dirichlet Allocation (LDA) [17], the Second Order Attributes (SOA) [18], the Document Occurrence Representation (DOR) [19], the Word2Vec [7] and the GloVe [20].

Villegas $et\ al.$ [21] compare these word embedding approaches for sentiment analysis by using several weighting schemes including $tf.idf$ and Boolean on a subset of the IMDB Review Dataset. They found out that the effectiveness of the LSA as the feature set with Naïve Bayes classifier outperforms other techniques [21]. In [22], Giatsoglou $et\ al.$ observed that LDA is computationally very expensive as compared to LSA on large data sets.

In [8], Mikolov $et\ al.$ argued that a high quality representation can be trained from huge data sets with billions of

words in the vocabulary. They developed a new model that preserved the linear regularities around terms and achieved high accuracy for vector operations. They found that neural networks performed better than LSA for preserving linear regularities around words. Then, a combination of sentiment lexicon and Word2Vec is investigated to add more features to the classification matrix in order to extract extra syntax and semantics feature from word. In contrast, Fan $et\ al.$ used Naïve Bayes as the classification method to build a sentiment lexicon through word vectors matrices separately, and then used the Boolean rules to classify the matched documents that appeared in both matrices for polarity [23].

Le $et\ al.$ proposed an approach based on the Paragraph Vector (Doc2Vec) for representing vectors as length of texts such as paragraph, sentence and documents [24] . In order to evaluate the effectiveness of the Doc2Vec, Lau $et\ al.$ used the Word2Vec with n-gram model to construct both Distributed Bag-of-Words version of Paragraph Vector (DBoW) and Distributed Memory version of Paragraph Vector (DMPV) for the Doc2Vec [25]. The results showed that for DBoW is better than DPMV model. Further analysis with different classifiers (SVM, Naïve Bayes and Maximum Entropy) showed that the unigram with SVM is the best [26].

For the dataset comprises of comments in Chinese for clothing products, a significant difference in the performance can be observed for Word2Vec with SVM perf classifier [27]. An extended model for sentiment classification based on the Paragraph Vector (Doc2Vec) [24], is also investigated by Haocheng $et\ al.$ in [28], which focused on the semantic features between words rather than the simple lexical or syntactic features. For micro-blog, Zhang $et\ al.$ investigated the use of multi-label classification, two micro-blog datasets, and eight different evaluation matrices on three different sentiment dictionaries [29]. In [30], the document vector was utilized to generate labeled dataset by using unsupervised learning approach through labeled training dataset.

III. FEATURE EXTRACTION METHOD BASED ON CLUSTERING FOR WORD2VEC

Constructing an effective features vector to represent text for classifier is an essential task in any text classification problem. In this paper, a method to construct the feature vector based on Word2Vec is proposed. The method consists of three main components, which are (1) the discovery of word embedding based on Word2Vec, (2) the clustering of terms in vocabulary based on opinion words, and (3) the construction of features matrix for classification based on cluster centroids as shown in Fig 1.

A. Learning Word Representation based on Word2Vec

The first component of the method deals with the discovery of word representation based on Word2Vec model. Given that a corpus D consists of a set of texts, $D = \{d_1, d_2, d_3, ..., d_n\}$, and a vocabulary $T = \{t_1, t_2, t_3, ..., t_m\}$, which consists of

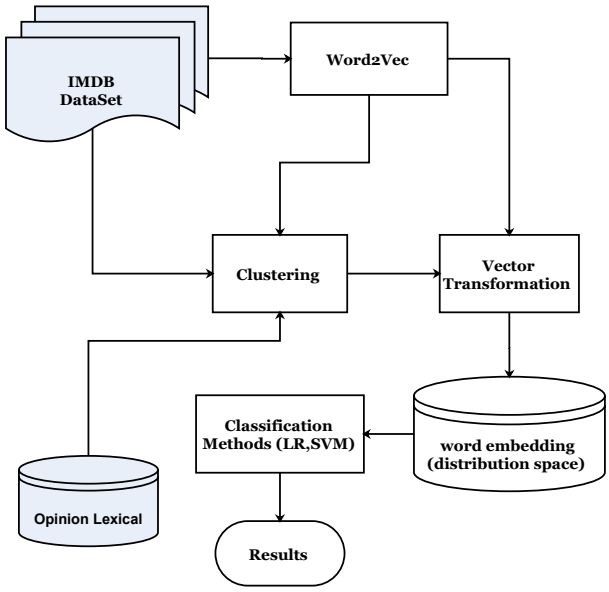

Fig. 1. Framework for the proposed method

unique terms extracted from D. The word representation of the terms t_i are discovered by using the Skip-gram model of the Word2Vec [31] to calculate the probability distribution of other terms in context given t_i. In particular, t_i is represented by a vector $\overrightarrow{v_i}$ that comprises of probabilistic values of all terms in the vocabulary. This word embedding technique is able to discover semantic relation among terms in the corpus. However, the resulting set of vectors for all terms in the corpus is high-dimensional and is inefficient for the classifier in the sentiment analysis task. As a result, this first component discovers a set of vector $V_T = \{\overrightarrow{v_1}, \overrightarrow{v_2}, \overrightarrow{v_3}, ..., \overrightarrow{v_m}\}$ representing the set of terms in the vocabulary T.

B. Clustering of Term Vectors based on Sentiment Lexical Dictionary

Constructing a feature matrix directly from the word representation produced by the Word2Vec model is inefficient as it tends to be huge due to high-dimensional nature of the word representation. In this paper, the terms t_i in the vocabulary are grouped into a set of clusters which eventually used to represent the text as a feature vector. As this paper focuses on the sentiment analysis, it is reasonable to assume that the clusters should be generated based on sentiment polarity of the terms in the vocabulary. However, most of the terms in the vocabulary are non-opinion words and do not have sentiment polarity. Therefore, all terms in T are clustered based on their similarity to the set of opinion words. As such, the polarity of non-opinion words is estimated based on the cluster they are in.

In order to cluster the terms based on polarity, a sentiment lexical dictionary that contains a list of opinion words (2005 of positive words and 4783 of negative words) is chosen as the centroid of the clusters. This dictionary has been proven to be

useful in many sentiment analysis techniques [32]. The aim is to group non-opinion words in the vocabulary into several clusters of opinion words obtained from the sentiment lexical dictionary. Let S be a set of opinion words (both negative and positive) from the sentiment lexical dictionary and C be the set of centroid terms, which consists of those common terms in the dictionary that are also appears in the vocabulary, $C = T \cap S$. Due to the curse of dimensionality problem of language model training, some of the terms in the sentiment lexical dictionary do not appear in the vocabulary of the Word2Vec. Thus, these words are ignored and are not used as the centroid. In this paper, almost 600 words from the dictionary are ignored.

As each terms in both the vocabulary $t_i \in T$ and the centroid $t_j \in C$ are represented by the vector discovered in the previous step, the similarity between both terms can be estimated based on the $cosine$ similarity of their respective vector representation such that, $sim(t_i, t_j) = cosine(\overrightarrow{v_i}, \overrightarrow{v_i})$. Therefore, for each term in T, its similatrity with all terms in C are calculated and the term is assigned to a cluster C_j if $sim(t_i, t_j)$ is the maximum for the term t_i and t_j is the centroid of the cluster C_j. As a result, all terms in the vocabulary are clustered based on the opinion words in the dictionary. Each cluster C_j will consist of a set of tuples representing the members of the cluster, such that each tuple consists of the term and its similarity to the centroid of the cluster, $\langle t_i, sim(t_i, t_j)\rangle$. Note that, since all the opinion words in the vocabulary T are used as the centroids, they are automatically part of a cluster and no tuples added for them.

C. Feature Extraction based on Polarity Clusters

The aim of the third component of the model is to contruct a feature vector for each text $d_i \in D$ based on the clusters discovered in the previous step. Therefore, instead of using the entire vocabulary size as the dimension of the feature vector, this approach will limit the dimension of the vector to the number of the cluster, $|C|$.

In order to construct the feature vector $\overrightarrow{vd_i}$ for a given text d_i, all the terms in d_i are scanned and the clusters that those terms belonged into are selected as the features to represent d_i. The weight is assigned to the column representing the cluster and is given by the similarity score between the centroid of the cluster and it's members. As there could be more than one member in the cluster, only a single value of similarity score is selected for the weight. In this paper, the maximum similarity score of the given cluster is selected as the weight for the feature. It has shown to be more effective than other scores, such as minimum or average.

Then, a feature matrix is constructed by combining all feature vectors for D into a matrix. As such, the resulting feature matrix of size $n \times |C|$ will be smaller than a typical feature matrix constructed based on bag-of-words, which is normally $n \times m$, due the dimension is limited to the number of clusters generated based on the proposed method. It is also expected that such reduction will have positive impact to the overall effectiveness of the sentiment analysis.

In addition, a simple transformation is applied to the feature matrix in order to improve the efficiency of the classifier. In particular, those similarity values belonged to the negative opinion words are changed by multiplying them to -1. As such, those columns with positive opinion words will have positive similarity values and those with negative opinion words will have negative similarity values. Such simple modification improves the speed of the classifier to almost 50%.

IV. Experimental Results and Analysis

In order to conduct the benchmark evaluation, the proposed method for sentiment analysis is evaluated by using the Large Movie Review Dataset (ACLIMDB), which is available online[1]. The dataset consists of 100,000 movie reviews and 50,000 of the reviews are labeled [33].

In this experiment, a standard sentiment lexical dictionary is used as the centroid for the clusters. The dictionary consists of 2005 positive terms and 4783 negative terms [11], [34]. The performance of the proposed method for sentiment analysis is compared based on the classification accuracy measure against the Word2Vec [9], the Doc2Vec [30] and the Bag-of-words [35] methods.

TABLE I shows that two classification techniques are used in this experiment, namely the Logistic Regression (LR) and the Support Vector Machine (SVM). Based on the table, it is obvious that the proposed method outperforms the other methods including the baseline Word2Vec for Logistic Regression. In addition, the accuracy of the proposed method is 93.80% as compared to 83.10% for the Word2Vec, which is an increase of 12.9%. In addition, the result is better than the other methods namely the Doc2vec (86.8%) and the BoW (89.15%). For the SVM classifier, the performance of the proposed method outperforms the baseline Word2Vec with an increase of almost 23.3%, from 70.25% to 86.6%. Its performance is comparable to the other two methods.

TABLE I
Accuracy of different feature sets for sentiment analysis

	Word2Vec	Doc2vec	BoW	Proposed method
Logistic Regression	83.10	86.80 (Δ4.5%)	89.15 (Δ7.3%)	93.80 (Δ12.9%)
Support Vector Machine	70.25	86.20 (Δ22.7%)	83.60 (Δ19.0%)	86.60 (Δ23.3%)

In addition, it is observed that the proposed method decreases the size of feature set to almost 80% of the Word2Vec size, which will reduce the complexity of the classifier. As a result, the proposed method will be more effective as well as efficient for sentiment analysis.

V. Conclusion

In this paper, a method is proposed to reduce the size of the Word2Vec feature set for sentiment analysis. The method

[1] http://ai.stanford.edu/~amaas/data/sentiment/aclImdb_v1.tar.gz

constructs cluster of terms centered by a set of opinion words from a sentiment lexical dictionary. A simple transformation is applied to the negative term vectors to redistribute the terms in the space based on their polarity. A much smaller matrix of document vectors is produced based on the set of clusters. Two classifiers, namely Logistic Regression and Support Vector Machine (SVM) are used to compare the performance of different feature set for sentiment analysis.

It has been observed that the performance of the proposed method is encouraging, showing that it can be more effective and efficient than the baseline. In the future, more investigation will be performed on the Word2Vec in term of the perplexity. In addition, another lexical dictionary with extend features will be used in clustering.

ACKNOWLEDGMENT

This work is partly supported by the Ministry of Higher Education Malaysia under the FRGS Grant (FRGS/1/2015/ICT04/UPM/02/5).

REFERENCES

[1] S. Shojaee and A. bin Azman, "An evaluation of factors affecting brand awareness in the context of social media in malaysia," *Asian Social Science*, vol. 9, no. 17, p. 72, 2013.

[2] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, C. Potts *et al.*, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, vol. 1631. Citeseer, 2013, p. 1642.

[3] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional LSTM networks for improved phoneme classification and recognition," in *International Conference on Artificial Neural Networks*. Springer, 2005, pp. 799–804.

[4] X. Lui and W. B. Croft, "Statistical Language Modeling For Information Retrieval," *Annual Review of Information Science and Technology 2005 Volume 39*, vol. 39, p. 1, 2003. [Online]. Available: http://ciir.cs.umass.edu/pubfiles/ir-318.pdf

[5] Z. Yu, H. Wang, X. Lin, and M. Wang, "Learning term embeddings for hypernymy identification." in *IJCAI*, 2015, pp. 1390–1397.

[6] X. Liu and W. B. Croft, "Statistical language modeling for information retrieval," *Annual Review of Information Science and Technology*, vol. 39, no. 1, pp. 1–31, 2005.

[7] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," *Nips*, pp. 1–9, 2013.

[9] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, pp. 1–12, 2013. [Online]. Available: http://arxiv.org/pdf/1301.3781v3.pdf

[10] C. C. Aggarwal and C. Zhai, *A Survey of Text Clustering Algorithms*, 2012.

[11] M. Hu and B. Liu, "Mining and summarizing customer reviews," *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining KDD 04*, vol. 04, p. 168, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1014052.1014073

[12] J. Kim, J. Yoo, H. Lim, H. Qiu, Z. Kozareva, and A. Galstyan, "Sentiment Prediction using Collaborative Filtering," *Icwsm*, 2013. [Online]. Available: http://www.isi.edu/{~}galstyan/papers/icwsm-CF.pdf

[13] R. Nikhil, N. Tikoo, S. Kurle, H. S. Pisupati, and G. R. Prasad, "A survey on text mining and sentiment analysis for unstructured web data," in *Journal of Emerging Technologies and Innovative Research*, vol. 2, no. 4 (April-2015). JETIR, 2015, pp. 1292–1296.

[14] H. Pouransari and S. Ghili, "Deep learning for sentiment analysis of movie reviews," 2014.

[15] E. Emad, E. M. Alshari, and H. M. Abdulkader, "Arabic Vector Space Model based on Semantic," in *International journal of computer science (IJISI),*, vol. 8, no. 6. Ain Shams, 2013, pp. 94–101.

[16] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge." *Psychological review*, vol. 104, no. 2, p. 211, 1997.

[17] M. Hoffman, F. R. Bach, and D. M. Blei, "Online learning for latent dirichlet allocation," in *advances in neural information processing systems*, 2010, pp. 856–864.

[18] A. P. López-Monroy, M. Montes-Y-Gomez, H. J. Escalante, L. V. Pineda, and E. Villatoro-Tello, "Inaoe's participation at pan'13: Author profiling task notebook for pan at clef 2013." in *CLEF (Working Notes)*, 2013.

[19] A. Lavelli, F. Sebastiani, and R. Zanoli, "Distributional term representations: an experimental comparison," in *Proceedings of the thirteenth ACM international conference on Information and knowledge management*. ACM, 2004, pp. 615–624.

[20] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation." in *EMNLP*, vol. 14, 2014, pp. 1532–1543.

[21] M. P. Villegas, M. José, G. Ucelay, J. P. Fernández, M. A. Álvarez-Carmona, M. L. Errecalde, and L. C. Cagnina, "Vector-based word representations for sentiment analysis: a comparative study," pp. 785–793.

[22] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas, "Sentiment analysis leveraging emotions and word embeddings," *Expert Systems with Applications*, vol. 69, pp. 214–224, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2016.10.043

[23] X. Fan, X. X. Li, F. Du, and X. X. Li, "Apply Word Vectors for Sentiment Analysis of APP Reviews," no. Icsai, pp. 1062–1066, 2016.

[24] Q. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," *International Conference on Machine Learning - ICML 2014*, vol. 32, pp. 1188–1196, 2014. [Online]. Available: http://arxiv.org/abs/1405.4053

[25] J. H. Lau and T. Baldwin, "An empirical evaluation of doc2vec with practical insights into document embedding generation," *arXiv preprint arXiv:1607.05368*, 2016.

[26] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Expert Systems with Applications*, vol. 57, pp. 117–126, 2016.

[27] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Predicting Valence-Arousal Ratings of Words using a Weighted Graph Method," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL-2015)*, pp. 788–793, 2015.

[28] H. Wu, Y. Hu, H. Li, and E. Chen, "A New Approach to Query Segmentation for Relevance Ranking in Web Search," *Inf. Retr.*, vol. 18, no. 1, pp. 26–50, 2015. [Online]. Available: http://dx.doi.org/10.1007/s10791-014-9246-7

[29] J. Zhang, C. Zong, and Others, "Deep Neural Networks in Machine Translation: An Overview," *IEEE Intelligent Systems*, vol. 15, 2015.

[30] S. Lee, X. Jin, and W. Kim, "Sentiment Classification for Unlabeled Dataset using Doc2Vec with JST," pp. 1–5, 2015.

[31] T. Mikolov, A. Joulin, S. Chopra, M. Mathieu, and M. Ranzato, "Learning Longer Memory in Recurrent Neural Networks," pp. 1–9, 2014. [Online]. Available: http://arxiv.org/abs/1412.7753

[32] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 168–177.

[33] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics, June 2011, pp. 142–150. [Online]. Available: http://www.aclweb.org/anthology/P11-1015

[34] B. Liu, M. Hu, and J. Cheng, "Opinion observer: analyzing and comparing opinions on the web," in *Proceedings of the 14th international conference on World Wide Web*. ACM, 2005, pp. 342–351.

[35] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning Word Vectors for Sentiment Analysis," *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, 2011.