International Journal of Future Generation Communication and Networking
Vol. 13, No. 2, 2020 pp.1530-1535

Improving the Accuracy of Sentiment Classification using Optimized Word Vector

M Gunasekar [1], K Kalaiarasan[2]
[1]guna18it@gmail.com, [2]kalaiarasank.it@mkce.ac.in

Abstract

Sentiment Analysis is emerging research area under Natural Language Processing (NLP). Sentiment Analysis has wide range of applications such as finding Customer opinion, digital marketing, politics etc., NLP tries to understand the meaning of the word from human perspective. Developing a Sentiment Classification model is a challenging task, as it involves lot of computation and memory requirement is high. Even then number of researchers has tried to simplify the model construction. There are numerous methods to capture the Sentiment from corpus, in which word embedding is showing better results. The word embedding models such as word2vec and Glove are widely used today. The disadvantage of the word embedding model does not perform well on small corpus. The proposed optimized word vector model improves the accuracy of the word embedding models. In our approach, we create vectors on the results of traditional Lexicon, POS tagging, word position algorithm and concatenate those vectors. The result shows that optimized word vectors has improved the accuracy of Sentiment classification.

Keywords: *Sentiment Analysis, Natural language processing, Word Embeddings, Deep Learning*

1. Introduction

Sentiment Analysis is an approach to identify the interest of the people on products, movies, politics etc., computing sentiment from a huge corpus is a complex process. There are numerous and tools available to capture the Sentiment from a document. The traditional methods of Sentiment classification are Lexicon based approach, using POS tagging, Statistical approach and deep Learning. Although these approaches use different strategies to capture sentiment from document, identifying the correct meaning of a word in a sentence is difficult. Researchers have worked on lot methods to improve the accuracy of sentiment classification. Word embeddings [2] are the pre-trained word vectors which works fine comparing to other traditional methods. The representation of the words from a document is important for Sentiment classification.

Semantic lexicon and corpus [3] are jointly used for word representation. The relation between the words are computed using the semantic lexicon and also identifies the co-occurrence of words. This incorporation method eases the word representation using sentiment lexicon process. In the distributed word to vector representation [4], the skip-gram model in word2vec is used to precisely capture the relationship among words. This approach also subsamples the more frequent words quickly and also shows that more phrases in the corpus can be easily identified. Input to Machine Learning models are fixed length feature vector [7], which reduces the efficiency of model since ordering and semantics are ignored. Paragraph vector representation learns the feature from the variable length texts from documents; paragraphs etc., and converts that to fixed length feature vector. Paragraph vector improves the text representation compared to legacy bag-of-words model.

2. Related work

A Deep contextual word vectors [8] are generated using the bidirectional language model. Bidirectional model consists of two layers and on top of this model Embeddings from Language model is built. ELMo model differs from traditional word embedding. Continuous space representation [9] uses the neural network to construct the word vectors. This model learns the relationship between the words automatically, for example King - Man + women compute a vector close to queen. This method answers the 40% of the analogy questions.

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC

Word Embedding:

The vector learning technique Glove [11] works on word co-occurrence to generate word vectors. From the co-occurrence matrix, the relationship between the words is identified and here the matrix constructed with the statistical information about words in corpus. The hybrid approach [13] on sentiment prediction combines the Word2Vec and sentiment lexicon method. The hybrid approach benefits in understanding the semantics of the documents much better. This approach works fine with most of the classifiers and better results are achieved with SVM classifier.

Sentiment Lexicon:

In vector representation statistical information is used to construct the word vector. Vector representation model may disregard the proper relation between the words. The vector space refining [12] model uses the relational information based on semantic lexicon to improve the vector representation. Semantic representation [14] method captures the relationship between words more precisely. The documents in the corpus are converted into word vector representation using word embedding algorithms, and then the semantic of the words in the vector space are defined using the lexicon based method. The extraction of semantic from the vector space further optimizes the results of the sentiment analysis. The unified model [15] for word representation works better than a single word vector representation. The word sense representation and word sense disambiguation complements one another to achieve a better word representation. This method eliminates the words with less importance from the word representation.

Deep Learning Method:

Deep learning [16] methods are helpful in doing the tasks in a hierarchical approach. The Deep learning models can work on both labeled and unlabelled data. The accuracy of the model relies on the hidden layers. There are different algorithms in deep learning which follows different strategy to implement the tasks. Distributed representations of word vectors [17] using feed forward and Recurrent Neural Net language model unveils the complex patterns between the words in the documents.

3. Proposed System

The works related to improve the accuracy of word vectors are being combined in our approach. In our work the different approaches such as Lexicon based method, PoS tagging, word position vector and other NLP techniques are combined to achieve optimized word vector. The optimized word vector architecture is being represented in Figure 1.

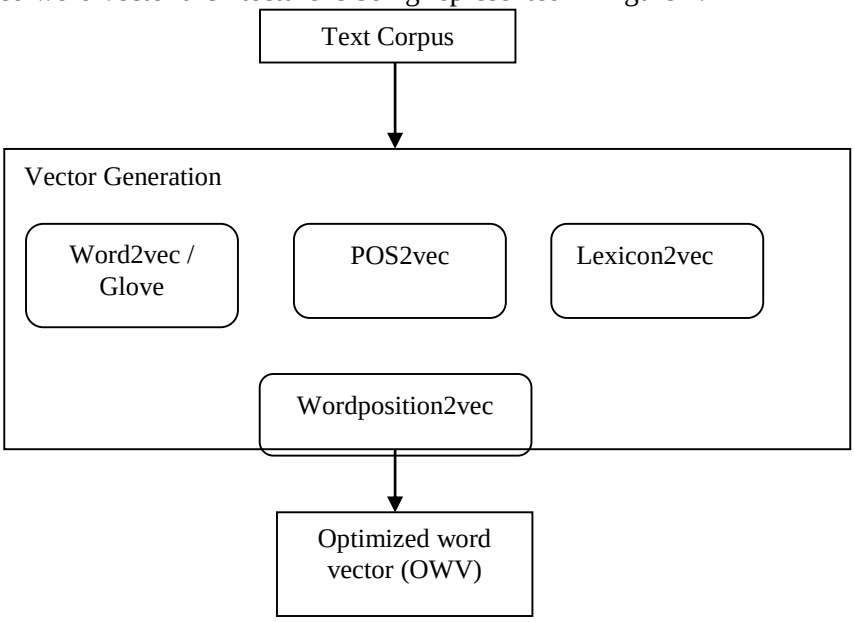

Figure 1: Optimized word vector architecture

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC

Lexicon based vector:

The lexicons are set of words with its polarity score values providing the insights of the word. Lexicon model consist of word and its sentiment values. There are lots of Sentiment lexicons available, choosing the lexicon having high impact for our problem is more important. We choose NRC Emoticon Affirmative Context Lexicon and NRC Emotion Lexicon, which groups the lexicons into 4 categories namely anger, fear, Joy, sadness.

Table 1: An Example of Lexicon based vector

S.no	Term	NRC Emoticon Affirmative Context Lexicon and NRC Emotion Lexicon
1	People	[0.121]
2	Spend	[0.297]
3	More	[-0.031]
4	Time	[0.234]
5	With	[-0.179]
6	Family	[0.515]

PoS Tagging:

The fundamental process in NLP is Parts-of-speech tagging. This method consigns a pos tag to every word in the document. This method carries more information about the neighbors of every word. PoS tag generated document is converted into vectors and combined with Glove vectors. Words with similar meaning will have the same vector. In figure 2 every word in the document (W_0 to W_5) is mapped with its corresponding PoS tag and converted into vectors.

Word Position Algorithm:

This algorithm calculates the relative distances between every word with other words in the document. The proximity of the words in the document provides a better knowledge to find the similarity between the words. In our example "people spend more time with family" the relative distance between more to spend and family are -1 and 4. In figure 2 depicts the relative distance between the words in the example and its corresponding vector representation.

	W_0	W_1	W_2	W_3	W_4	W_5
Words [	People	spend	more	time	with	family
PoS tag [	<NNS>	<VB>	<AVB>	<NNS>	<WITH>	<NNS>
Vectors [	d_{NNS}	d_{VB}	d_{AVB}	d_{VB}	d_{WITH}	d_{NNS}
Rel.distance [	[0,5]	[-1,4]	[-2,3]	[-3,2]	[-4,1]	[-5,0]
Vector [	$P_0=[d_0,d_5]$	$P_1=[d_{-1},d_4]$	$P_2=[d_{-2},d_3]$	$P_3=[d_{-3},d_2]$	$P_4=[d_{-4},d_1]$	$P_5=[d_{-5},d_0]$

Figure 2: An example of PoS tagging and Position to vector

Word2Vector Model and Glove:

Word2vec provides the good quality word embeddings constructed based on CBOW and Skip-gram model. The vector generated using the word2vec model contains the words in the close proximity in the document. Our method takes a sentence form the corpus and returns

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC

optimized word vector of the sentence. If the input sentence is "people spend more time with family", then input is divided into set of words W ={ people, spend, more, time, with, family. In the next step POS tags are assigned to the words and constant vector is generated as the result of second phase. In the third step the word2vec method is used to generate the word vector for every word in the sentence. If the word is not present in the NRC lexicon then its vector is assigned as zero. Finally all the vectors from the previous steps are concatenated. The optimized word vector representation is as follows,

$OWV_{people} = (d_{People}, d_{NNS}, P_0, d_{lex\text{-}people})$,
$OWV_{spend} = (d_{spend}, d_{VB}, P_1, d_{lex\text{-}spend})$,
$OWV_{more} = (d_{more}, d_{AVB}, P_2, d_{lex\text{-}more})$,
$OWV_{time} = (d_{time}, d_{NNS}, P_3, d_{lex\text{-}time})$,
$OWV_{with} = (d_{with}, d_{NNS}, P_4, d_{lex\text{-}with})$,
$OWV_{family} = (d_{family}, d_{NNS}, P_5, d_{lex\text{-}family})$,
OWV_{people}, d_{people}, d_{NNS}, P_0 and $d_{lex\text{-}people}$ represents the optimized word vector of people.

Dataset:

We use IMDB and Rotten tomatoes movie review dataset prepared by Pang and Lee, which contains 2000 and 10662 reviews respectively. Both the dataset contains equal number of positive and negative reviews.

4. Results

We used two sentiment lexicons and bigram score in our approach. The Lexicon scores are normalized. Our method is tested on two different deep learning models.

Table 2: Results of our approach

Model	Dataset	Word2Vec	Glove	OWV
Model 1 (static)	IMDB	80.2	79.7	81.5
	RT	80.5	80.1	81.9
Model 2 (non-static)	IMDB	80.4	80.1	81.7
	RT	80.9	80.3	82.1

The dataset is tested under two different models and the proposed approach shows better results compared to the existing word embedding models. The optimized word vector outperforms the Word2vec and Glove methods in both the conditions.

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC

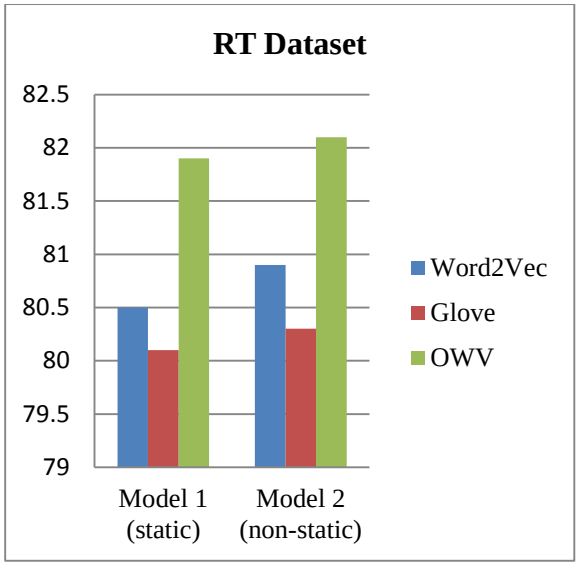

Figure 3: Performance of RT dataset

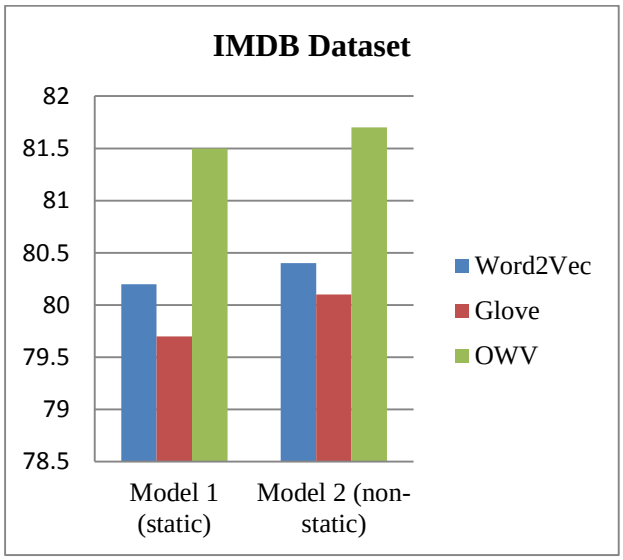

Figure 3: Performance of IMDB dataset

The above chart shows that the proposed optimized word vector method has high accuracy than the word2vec and Glove methods.

5. Conclusion

In this paper, we created the vectors for every model. Then the vectors are concatenated to form the final word vector. The performance of the two well known word embedding methods is compared with the proposed method OWV. The results show that the proposed method outperforms the traditional methods. The proposed method has improved the accuracy of sentiment identification task under all models. The accuracy of the pre-trained vectors can be increased by concatenating with other vectors. The correct selection of Lexicon method also improves the accuracy of the sentiment analysis technique. Our proposed approach increases the accuracy of the sentiment classification technique

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC

References

[1] Wenqian Shang, H Huang, H Zhu, Y Lin "A novel feature selection algorithm for text categorization" Expert Systems with Applications 33.1 (2007): 1-5.
[2] Rezaeinia, Seyed Mahdi, Ali Ghodsi, Hadi Veisi "Sentiment analysis based on improved pre-trained word embeddings." Expert Systems with Applications 117 (2019): 139-147.
[3] Bollegala, Danushka, T Maehara "Joint word representation learning using a corpus and a semantic lexicon." Thirtieth AAAI Conference on Artificial Intelligence. 2016.
[4] Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. "Distributed representations of words and phrases and their compositionality." In Advances in neural information processing systems, pp. 3111-3119. 2013.
[5] Mesnil, Grégoire, MA Ranzato, Y Bengio "Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews." arXiv preprint arXiv:1412.5335 (2014).
[6] Agarwal, Basant, and Namita Mittal. "Optimal feature selection for sentiment analysis." International Conference on Intelligent Text Processing and Computational Linguistics. Springer, Berlin, Heidelberg, 2013.
[7] Le, Quoc, and Tomas Mikolov. "Distributed representations of sentences and documents." In International conference on machine learning, (2014) pp. 1188-1196.
[8] Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. arXiv preprint arXiv:1802.05365.
[9] Mikolov, Tomáš, Wen-tau Yih, and Geoffrey Zweig. "Linguistic regularities in continuous space word representations." In Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies, pp. 746-751. 2013.
[10] Mikolov, T.; tau Yih, W.; and Zweig, "Linguistic regularities in continous space word representations." In Proc. (2013) of NAACL, 746 – 751.
[11] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.
[12] Faruqui, M., Dodge, J., Jauhar, S. K., Dyer, C., Hovy, E., & Smith, N. A. (2014). Retrofitting word vectors to semantic lexicons. arXiv preprint arXiv:1411.4166.
[13] Giatsoglou, Maria, Manolis G. Vozalis, Konstantinos Diamantaras, Athena Vakali, George Sarigiannidis, and Konstantinos Ch Chatzisavvas. "Sentiment analysis leveraging emotions and word embeddings." Expert Systems with Applications 69 (2017): 214-224.
[14] Bullinaria, John A., and Joseph P. Levy. "Extracting semantic representations from word co-occurrence statistics: A computational study." Behavior research methods 39, no. 3 (2007): 510-526.
[15] Chen, Xinxiong, Zhiyuan Liu, and Maosong Sun. "A unified model for word sense representation and disambiguation." In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1025-1035. 2014.
[16] Young, Tom, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. "Recent trends in deep learning based natural language processing." ieee Computational intelligenCe magazine 13, no. 3 (2018): 55-75.
[17] Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient estimation of word representations in vector space." arXiv preprint (2013) arXiv:1301.3781.

ISSN: 2233-7857 IJFGCN
Copyright ⓒ2020 SERSC