SENTIMENT ANALYSIS OF ONLINE PRODUCT REVIEWS WITH WORD2VEC N-GRAMS

[1] Poonam Choudhari,[2] Dr.S.Veenadhari

[1]Research Scholar,[2]Associate Professor
[1] Department of Computer Science & Engineering ,
[1] Rabindranath Tagore University ,Bhopal, India

Abstract : Sentiment Analysis is a process to understand the feeling/attitude/sentiment towards a written piece of text by analyzing and then classifying the text as positive, negative or neutral. Data is one of the important aspect that should be handled and represented carefully in the classification process, which affects the performance of the classifier. In sentiment classification process data is represented as feature vector. Our work focuses on investigating the performance of Word2Vec N-grams (Unigrams and Bigrams) feature vector in the sentiment classification process of online consumer reviews about different products. The feature vector are tested on different well known text classifiers used for Sentiment Analysis .We also investigated the performance of feature vector as the size of dataset is increased . Thus it would help to find optimal combination which could enhance the performance of classifier.

***IndexTerms* - Sentiment Analysis ,Word2Vec ,CBOW, Skip-Gram, Random Forest, LRCV,MLP, Naïve Bayes, Decision Tree**

I. INTRODUCTION

In today's digital era , people frequently use digital platform to express opinions or read opinions about some event, person ,product or entity so a large number of product reviews are posted daily on online platforms. One of the major challenge that is been faced is to analyze those unstructured reviews and find the customer perspective that whether the product has a positive, negative or neutral impact on their mind. Sentiment Analysis [1] is a process to analyze such opinions/ reviews and classify them into positive, negative or neutral category. Sentiment classification can be done by Lexicon Based ,Machine Learning Based or Hybrid of two .In our approach we have used Machine learning Based Sentiment Classification[3]. Feature Vector i.e. data representation is an important part of sentiment classification[2]. The performance of classification can be improved if efficient feature vectors are used. Here we have used an efficient feature vector Word2Vec which is different from traditional feature vectors like Bag- of- Words, TF-IDF . Word2Vec preserves the semantic meaning between the words of the corpus. We have investigated two models of Word2vec i.e. Continuous- Bag of words(CBOW) and Skip-gram. Both unigrams(single word) and Bigrams(sequence of two words) of models are tested to evaluate the performance of classification process. The classifiers used are Logistic Regression CV, Multilayer Perceptron, Random Forest, Decision Tree and Gaussian Naïve Bayes. Here we have used Amazon Mobile product reviews database for classification and tested the performance of feature vector as the dataset size is increased. When evaluating the feature vector for product review based sentiment analysis, we are primarily concerned with determining the best Accuracy score and F1-score for sentiment classification.

The paper is organized as follows. Section II gives a literature survey on the work done by various researchers related to sentiment analysis. Section III describes the methodology used for classification and brief explanation of Word2vec N-gram feature vector. Section IV describes experimental results, concludes the paper and discusses the future work.

II. LITERATURE SURVEY

This section gives a brief overview of the research work done in the field of Sentiment Analysis with Word2vec Feature Vector.
In this paper[4] by Barkha Bansal et al., Word2Vec model is used as feature vector and applied on the mobile phones dataset taken from Amazon. They have used CBOW and skip-gram models of Word2Vec with different machine learning algorithms like SVM, Naïve Bayes, Logistic Regression and Random Forest. The combination of CBOW with Random forest give the best result .
In this paper [5] by Marwa Naili et al., the author investigated performance of word embedding in the field of topic segmentation here the document is partitioned into segments so that each segment represent some topic. The feature vector used are- Word2Vec,Glove, and LSA. In case of Word2vec both CBOW and Skip-gram models are used with hierarchical softmax and negative sampling algorithm. The datasets used are in English and Arabic language. The quality of topic segmentation depends on the language used and is better in case of English language than Arabic language due to its complexity. Independent of the language used, negative sampling give the best result.Word2vec performed well among the three feature vectors. CBOW give better result with frequent words while skip gram give better results with infrequent words.
In the paper[6] by Joshua Acosta et al., the author has done sentiment analysis of twitter data related to U.S. Airlines. The proposed work is executed by taking CBOW and Skip-gram models of Word2vec feature vector . The classifier used are Gaussian Naïve Bayes, Bernoulli Naïve Bayes, Logistic Regression SVM. The combination of Skip-gram with Logistic Regression and SVM give the best result .

In this paper[7] by Eissa M.Alshari et al. , a solution is proposed to deal with sentiment analysis having corpus which are not balanced. First a number of clusters are formed for majority class samples. The number of elements are calculated in each of the group and filter the elements which are similar by comparing with other elements. All group elements are then gathered together to form majority class training set and same number of minority elements are taken to get a balanced corpus. Another method is also employed to select the group of elements in the cluster by choosing the elements near to the centroid so as to have a balanced distribution of features. The proposed method performed well as compare to existing balancing techniques.
In this paper[8] by Sadam Al-Azani et al., sentiment analysis is done on highly imbalanced data in Arabic language. SMOTE sampling technique is used for balancing the majority and minority instances.Word2vec CBOW model is used for feature vector generation The classification performance is tested on various base classifiers and their ensembles.Word2vec with SMOTE and ensemble classifier achieved 15% better F1 score over base classifier without SMOTE.

III. Data And Methodology

In this section first a brief description about the dataset used for the proposed methodology is described After that the steps of the proposed methodology are explained. As we are investigating the performance of Word2vec feature vector ,a brief description of the Word2vec feature vector is also explained. Our method consists of following steps :

- Data Pre-processing
- Word Vector Representation of Pre-processed data.
- Sentiment Classification

3.1 Data Description

The corpus is taken from the Amazon Unlocked Mobile Phone Reviews publicly available on Kaggle[9] which consists of online customer reviews of mobile phones sold on Amazon. The fields of the dataset consists of Product Name, Brand Name, Price, Rating, Reviews and Review votes .For the analysis purpose ,we took only reviews and rating field from the dataset. The reviews are divided into positive and negative sentiment according to their rating given by customer. The rating which consists of four and five are labeled as positive sentiment and rating with one and two are labeled as negative sentiment. The dataset is unbalanced as it consists of more positive reviews and only less negative reviews. So the dataset is balanced by applying SMOTE (Synthetic Minority Over Sampling Technique)[10]data sampling technique after feature vector conversion. We have taken different dataset size for classification. As we are doing Machine leaning based classification the dataset is divided into training and testing set in the ratio of 70:30 .The classifier is trained on the training set and then performance is evaluated on the testing dataset.

3.2 Proposed Methodology

The proposed methodology in implemented in python language .We have used well known libraries of python for implementing the various steps of our proposed method. The steps of the proposed methodology are described below.

3.2.1.Data Pre-processing and Cleaning

In the first step the raw data is pre-processed and cleaned reviews are obtained .It is done by removing unwanted digits ,symbols, HTML tags. Conversion of all the words to lower case .Stemming i.e. conversion of word to their root form is done by using Snowball stemmer. Filtering of stop words are also performed by using English stop word list of NLTK.

3.2.2.Word Vector Representation of Pre-processed data

In the second step pre-processed data is converted into feature vector .It means that the data is tokenized into words and converted to numerical representation so that it can be understood by the machine learning classifier. In our methodology,Word2Vec is used as feature vector due to its unique qualities that makes it different and efficient as compared to traditional feature vector representations like Bag-of –Words and TF-IDF. A brief description of the feature vector is explained below.

Word2vec[11] is a shallow neural network feature vector representation technique that produces word embedding which captures semantic relationship between the words. The technique was developed by Tomas Mikolov[12] at Google in 2013. It consist of two layer neural network where there is a input ,one hidden layer and output layer .The input layer consists of the words tokenized in data corpus. Punkt tokenizer of NLTK is used for tokenization of data corpus. The output layer consists of the corresponding feature vector for the tokenized words in the data corpus. It creates the vectors that are distributed numerical form of the words.
One of the qualities of Word2vec is that it group the vector with similar context together in vector space. It calculates the cosine similarity distance between the words to find similar context. Cosine similarity with zero degree angle is equal to one means it is the exact word that is taken into consideration i.e. battery equals battery. Cosine similarity with ninety degree angle means no similarity between the words. For e.g. in our dataset that is related to mobile domain if some major features of mobile are considered ,then following feature vector with similar context are obtained.

Table 1 : Mobile phone Features with their respective feature vector with similar context obtained by Word2vec

Feature	Feature Vectors with similar context
'battery'	'lasts', 'hours', 'night', 'charge', 'day'
'camera'	'flash', 'pics', 'front', 'pictures', 'color'
'price'	'value', 'deal', 'quite', 'budget', 'beat'
'screen'	'touch', 'keyboard', 'fingers', 'cover', 'clear'
'size'	'small', 'large', 'speed', 'smart phones', 'inch'
'sound'	'loud', 'speaker', 'speakers', 'music', 'headphone'

There are two models of Word2vec. They are CBOW(Continuous -Bag Of- Words) and Skip-gram. In our methodology both models are used with their N-grams i.e. unigrams (single word) and bigrams(two –word sequence of words) are evaluated in our methodology

3.2.2.1.CBOW((Continuous Bag -Of -Words)

This technique involves predicting the target word using the related context words.

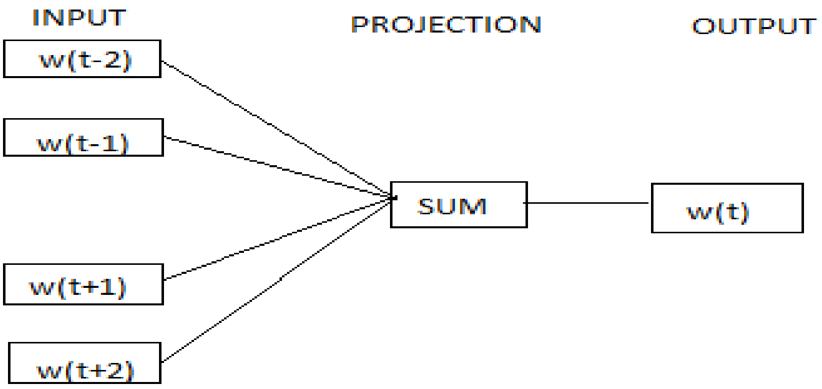
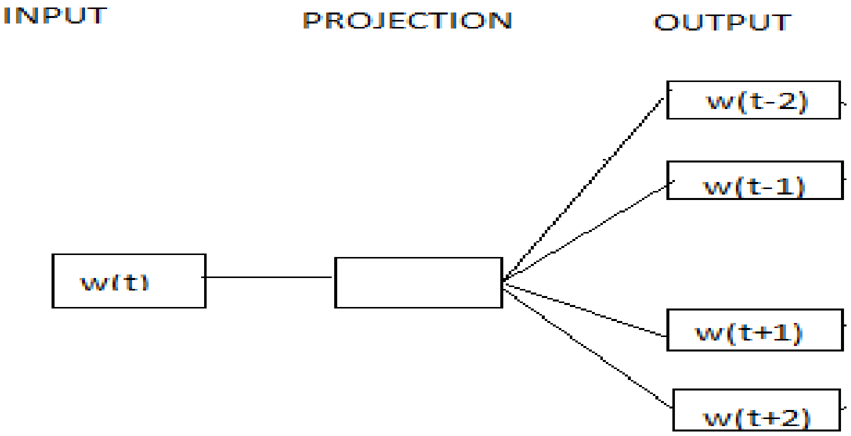


Figure 1: CBOW model of Word2vec

3.2.2.2.Skip-gram

This technique is opposite of CBOW which involves predicting the target context words using the related word.



.

Figure 2:Skip-gram model of Word2vec

The feature vectors are obtained by using Gensim library[13] of python. Some hyper parameters need to set to obtain the word vectors. We have used the following hyper parameters

- Number of features = 300
- Minimum word count = 10
- Number of workers = 4
- Window Size = 10

- Down sampling = 1e-3

Number of features represents the embedding dimension size of feature vector. Minimum word count represents that words considered for vectorization will not be lower than this frequency. Number of workers represents the number of threads to train the model. Window size means the maximum size that will be considered between the current word and the word to be predicted in the sentence. Down sampling represents the threshold for down sampling the words with higher frequency.

3.2.3.Sentiment Classification

This is third and final step of our proposed method. The training and testing set of word vectors are obtained from the previous step. Next the machine learning based classifiers are trained on the training dataset. After that the classifier are evaluated on the testing dataset. The classifiers used are Logistic Regression CV,MLP(Multi Layer Perceptron), Random Forest, Decision Tree and Gaussian Naïve Bayes[14][15]. The classifiers used are implemented using Scikit learn package of python.

IV. Results and Discussion

Our experiment consists of two parts .One is classification of product reviews into negative and positive category by using the two models of Word2vec i.e. CBOW and Skip-gram method with their unigrams and bigrams. Second the classification is done on different number of reviews i.e. gradually increasing the number of reviews in the dataset. We have taken the following number of reviews : 6,655 reviews,16,678 reviews , 26,764 reviews and 40,227 reviews in the dataset. The metrics used for evaluation are Accuracy and F1 score . Accuracy is the ratio of number of correct predictions to the total number of predictions. It predicts the part of prediction our classification model predicted right. F1 score represents the harmonic mean of precision and recall .It selects a classification model on the basis of balance between precision and recall. Precision is the number of True Positive divided by the sum of True Positive and False Positive. Recall is the number of True Positive divided by the sum of True Positive and False Negative.

Accuracy = (True Positive +True Negative) /(True Positive +True Negative+ False Positive +False Negative)

F1 score = 2*(Precision *Recall) / (Precision + Recall)

Table 2 : Comparison of Accuracy and F1- score on different classifiers with different dataset size

Number of Reviews in the Dataset	**Classifier**	**CBOW(Unigrams)**		**CBOW(Uni +Bigrams)**		**Skip-gram(Unigrams)**		**Skip-gram(Uni +Bigrams)**	
		Accuracy	**F1-Score**	**Accuracy**	**F1-Score**	**Accuracy**	**F1-Score**	**Accuracy**	**F1-Score**
6655	**MLP**	0.808713	0.849488	0.792188	0.834727	0.810215	0.849663	0.803205	0.846544
	LRCV	**0.819229**	**0.861314**	**0.812719**	**0.855263**	**0.824737**	**0.866003**	**0.826740**	**0.866615**
	RF	0.794692	0.838455	0.805709	0.851227	0.753130	0.802088	0.704056	0.751367
	DT	0.754632	0.803055	0.799199	0.849756	0.708563	0.756485	0.661492	0.698752
	GNB	0.621432	0.655738	0.665999	0.707584	0.571357	0.597744	0.553330	0.576046
16678									
	MLP	0.784572	0.825510	0.788569	0.828747	0.765987	0.805190	0.712030	0.744549
	LRCV	**0.844924**	**0.881091**	**0.837730**	**0.874730**	**0.842326**	**0.878485**	**0.840727**	**0.876911**
	RF	0.784772	0.831112	0.799161	0.843580	0.757994	0.806085	0.738209	0.789321
	DT	0.767986	0.822123	0.740807	0.789482	0.743805	0.798110	0.709233	0.764677
	GNB	0.603717	0.638469	0.611511	0.646031	0.585132	0.618523	0.562150	0.590850
26764									
	MLP	0.837983	0.875108	0.801494	0.840088	0.815567	0.853468	0.781320	0.819638
	LRCV	**0.850560**	**0.885584**	**0.838356**	**0.875456**	**0.850311**	**0.885349**	**0.841843**	**0.878213**
	RF	0.785181	0.830600	0.784433	0.830576	0.780448	0.827411	0.731258	0.779481
	DT	0.739975	0.788364	0.768742	0.821253	0.748319	0.798283	0.720672	0.779167
	GNB	0.610336	0.642849	0.601494	0.633280	0.600623	0.633108	0.573350	0.600420
40227	**MLP**	0.848123	0.884768	0.817135	0.856437	0.813903	0.851749	0.819869	0.858739
	LRCV	**0.852846**	**0.888105**	**0.848206**	**0.883948**	**0.851852**	**0.887064**	**0.851106**	**0.886775**
	RF	0.787223	0.833398	0.787555	0.834538	0.768001	0.814594	0.740244	0.790118
	DT	0.773469	0.823271	0.750352	0.802335	0.751346	0.799358	0.701135	0.752962
	GNB	0.603778	0.638221	0.591515	0.623836	0.596404	0.632294	0.576187	0.607775

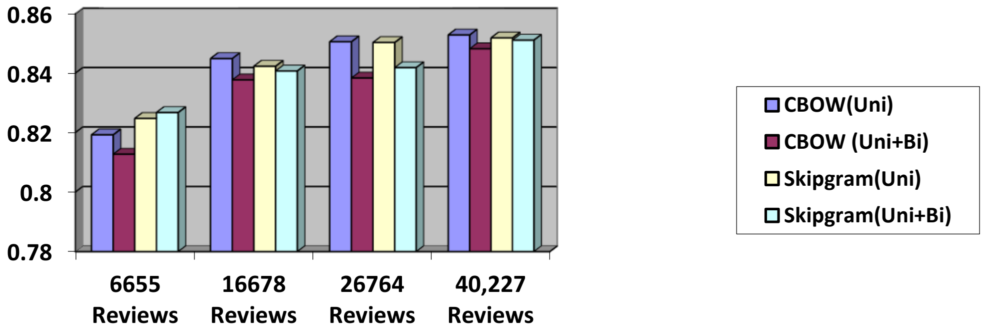


Figure 3 : Comparison Chart showing the accuracy scores with LRCV classifier

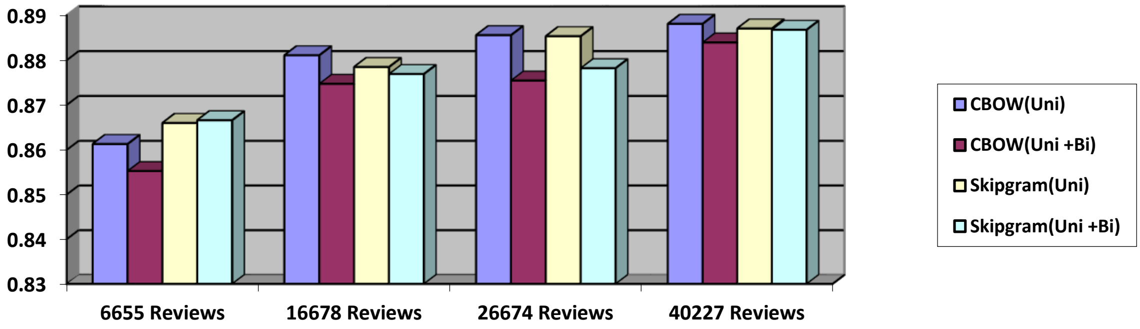


Figure 4 : Comparison Chart Showing the F1 –Scores with LRCV Classifier

The table above shows the results obtained by applying different machine learning classifiers with different dataset size on Word2vec N-grams. The highest accuracy and f1- score obtained with the different dataset size are given in bold text. Logistic Regression CV classifier gives the best score in terms of accuracy and F1-Score for all the dataset size. The Comparison chart shows the accuracy and F1-score with LRCV classifier. It shows that the best accuracy score and F1 score are obtained with CBOW Unigrams model . It can be seen from the results that as the number of reviews increases in the dataset the feature vector preformed well. The best scores are obtained with the highest number of reviews i.e. with 40,227 reviews. In future, other feature vectors like Glove and Fast Text with other machine learning classifiers or their hybrid can be used to evaluate their performance.

REFERENCES

[1] Bing Liu. 2012.Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers.

[2] Bo Pang and Lillian Lee .2008.Opinion Mining and Sentiment Analysis , Foundations and Trends_ in Information Retrieval. Vol. 2, Nos. 1–2 DOI:10.1561/1500000001, 1–135.

[3]B Pang B.,Lee , and L.,Vaithyanathan.2002.Thumbs up? Sentiment Classification using Machine Learning Techniques. Association for Computational Linguistics, Proceedings of the conference on Empirical Methods in Natural Language Processing, pp. 79–86.

[4] Barkha Bansal,Sangeet Shrivastava .2018.Sentiment Classification of online consumer reviews using word vector representations. International Conference on Computational Intelligence and Data Science .Elsevier, Science Direct, Procedia Computer Science 132 ,1147-1153.

[5] Marwa Naili, Anja Habacha Chaibi, Henda Hajjami Ben Ghezala.2017. Comparative study of word embedding methods in topic segmentation. International Conference on Knowledge Based Intelligent Information and Engineering Systems, Elsevier, Science Direct, Procedia Computer Science 112 ,340-349.

[6] Joshua Acosta et al.2017.Sentiment Analysis of Twitter Messages using Word2Vec. Proceedings of Student-Faculty Research Day, CSIS, Pace University, Pleasantville ,New York .

[7] Eissa M.Alshari et al.2017.Improvement of Sentiment Analysis based on Clustering of Word2Vec Features. 28th InternationalWorkshop on Database and Expert Systems Applications, IEEE doi 10.1109/dexa.2017.41.

[8] Sadam Al-Azani, and El-Sayed M. El-Alfy.2017.Using Word Embedding and Ensemble Learning for Highly Imbalanced Data Sentiment Analysis in Short Arabic Text. The 8th International Conference on Ambient Systems, Networks and Technologies, ANT.

[9] https://www.kaggle.com/PromptCloudHQ/amazon-reviews-unlocked-mobile-phones/.

[10]NiteshV.Chawlaetal.2002.SMOTE:synthetic minority oversampling technique.Journalof Artificial IntelligenceResearch,pages321–35.

[11] Tomas Mikolov et al. 2013.Efficient Estimation of Word Representations in Vector Space . arXiv:1301.3781v3[cs.CL].

[12]Tomas Mikolov et al.2013. Distributed Representations of Words and Phrases and their Compositionality.arXiv:1310.4546v1[cs.CL] .

[13] https://radimrehurek.com/gensim/models/word2vec.html.

[14] Vivek Narayanan, Ishan Arora, Arjun Bhatia .2013.Fast and accurate sentiment classification using an enhanced Naive Bayes model. International Conference on Intelligent Data Engineering and Automated Learning ,Springer .

[15] Mayy M. Al-Tahrawi.2015.Arabic Text Categorization Using Logistic Regression .I.J. Intelligent Systems and Applications,06, 71-78.