Sentiment Analysis Of Car Brand Reviews Dataset With Word2Vec N Grams

Poonam Choudhari [1], Dr.S.Veenadhari [2]
[1] Rabindranath Tagore University, Bhopal, India
[2] Rabindranath Tagore University, Bhopal, India
(E-mail: choudhari.poonam@gmail.com)

Abstract— Sentiment Analysis is a technique to find the feeling or sentiment expressed in a written piece of text by classifying the text as positive, negative or neutral. One of the important tasks in the sentiment classification process is the representation of data that is done by feature vector. The paper concentrates to inspect the performance of Word2Vec N grams i.e. Unigrams and Unigrams plus Bigrams (with phrases) feature vector in the sentiment classification process related to consumer reviews about different car brands. The feature vector are experimented with different well known machine learning classifiers used for Sentiment Analysis to find which classifier gives the highest scores in terms of Accuracy and F1 Score. The performance of feature vector is also analyzed as the size of dataset is increased.

Keywords— Sentiment Analysis; Word2Vec ;CBOW; Skip-gram; Random Forest; LRCV; MLP ;Naïve Bayes; Decision Tree

I. INTRODUCTION

In today's competitive business world, companies want to find out what consumers think about their brand. The consumers frequently use digital platform to express their opinions about the product brands they have used. The consumer's perspective is useful for both manufacturers and the future consumers that are thinking to buy the product. The major issue that is been faced is to analyze those thousands of unstructured reviews manually and to gain a knowledge about consumer's view that whether the brand has a positive, negative or neutral impact on their mind. Sentiment Analysis [1] is an automated process to understand this business intelligence by analysing such opinions/ reviews and classify them into positive, negative or neutral category. The techniques used to perform Sentiment Analysis are Lexicon Based, Machine Learning Based or Hybrid of two .In the paper we have used Machine learning Based approach for Sentiment Classification [3]. Feature Vector i.e. data representation is an important task of sentiment classification [2]. A good feature vector that represents valuable information about the data can help to better classify the data. A context based feature vector Word2Vec is used in our approach which is more efficient from traditional feature vectors like Bag- of- Words, TF-IDF. Word2Vec preserves the semantic meaning between the features of the corpus reviews. We have inspected two models of Word2Vec i.e. Continuous- Bag of words (CBOW) and Skip-gram. Unigrams and Unigrams + Bigrams (with Phrases) are tested with both the models to evaluate the performance of classification process. The unigrams refers to single word whereas Bigrams refers to sequence of two words. The classifiers used for sentiment classification are Logistic Regression CV, Multilayer Perceptron, Random Forest, Decision Tree and Gaussian Naïve Bayes. Here we have used consumer reviews database about different car brands for classification and tested the performance of feature vector with different machine learning classifiers as the dataset size is increased. When evaluating the feature vector for car domain related review based sentiment analysis, we are primarily concerned with determining the best Accuracy Score and F1-Score for sentiment classification.

The organization of the paper is as follows. Section II gives a literature review of the work done previously by various researchers related to sentiment analysis with Word2Vec. Section III describes the methodology used for classification and brief explanation of Word2Vec with N-gram (phrases) feature vector. Section IV describes experimental results, concludes the paper and discusses the future work.

II. LITERATURE SURVEY

A brief overview of the research work done in the field of Sentiment Analysis with Word2vec Feature Vector is given in this section.

Barkha Bansal et al. [4] applied Word2vec feature vector on mobile phones dataset taken from Amazon. The Author used both CBOW and Skip gram models with well-known classifiers like SVM, Naïve Bayes, Logistic Regression and Random Forest. They experimented with different dimensions of window size of Word2vec and found improvement with increasing dimensions. They found the combination of CBOW with Random forest classifier has given the best score.

Marwa Naili et al.[5] inspected the performance of word embedding in the field of topic segmentation The document that is used as input is divided into segments where each segment represent some topic. The author used Word2Vec,Glove, and Latent Semantic Analysis(LSA) feature vector for comparison. Both Skip-gram and CBOW models of Word2Vec are used with hierarchical softmax and negative sampling algorithm. They experimented with both English and Arabic languages. CBOW gave better result with frequent words while skip gram gave better results with infrequent words. 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 gave the best result.Word2Vec and Glove performed well in both English and Arabic language in comparison to LSA.Word2vec gave the best feature vector representation.

Joshua Acosta et al.[6] has performed sentiment analysis of twitter data related to U.S. Airlines. The author used both CBOW and Skip-gram models of Word2vec feature vector. The classifiers used are Gaussian Naïve Bayes, Bernoulli Naïve Bayes, and Logistic Regression. SVM and LR both with skip gram model produce the best accuracy scores among all.

Eissa M.Alshari et al.[7] proposed Word2Vec feature vector with low dimensions. They have done clustering of word vectors obtained based on the opinion words of sentiment dictionary. They experimented with Logistic regression and SVM as machine learning classifiers with IMDB dataset. They compared their work with simple Word2Vec, Doc2Vec and Bag of Words model. They found that the proposed feature vector performed well in comparison to other feature vectors. Logistic Regression performed well in comparison to SVM.

Sadam Al-Azani et al. [8] experimented with highly imbalanced data in Arabic language for performing sentiment analysis. A data sampling technique known as SMOTE is used for balancing the database so as to make the majority and minority classes equal to each other. CBOW model of Word2Vec is used for feature vector generation. The classification performance is tested on various base classifiers and their ensembles. The feature vector when applied with SMOTE and ensemble classifier achieved 15% better F1 score on average over base classifier.

In the above papers, the N-grams(phrases) of the feature vector Word2vec has not been used, so in this paper we have investigated the Word2Vec-Unigrams and Unigrams + Bigrams (with phrases) to know its performance with various machine learning classifiers.

III. DATA AND METHODOLOGY

In this section, a brief explanation about the dataset used for the proposed methodology is given. The steps of the methodology are explained. As we are inspecting the performance of Word2Vec feature vector, a brief description about Word2Vec feature vector is also explained. Our method consists of following steps:

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

A. Data Description

The dataset is taken from Car Domain publicly available on Kaggle [9] which consists of online customer reviews about different car brands. The columns of the dataset consists of Vehicle Title, Author name, Rating, Reviews, Review Title and Review Date .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 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 machine learning based classification is used in our approach, 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.

B. Proposed Methodology

The implementation of the technique is done in python language .We have used well known packages of python for implementing the various steps of our method. The steps of the methodology are explained below.

- Data Pre-processing

The first step consists of getting the cleaned data from the raw data by pre processing the raw data. Unwanted digits, symbols, HTML tags that do not contribute in classification process are removed from the reviews. Conversion of all the words from upper case to lower case .The process of stemming i.e. conversion of word to their root form is done by using Snowball stemmer. Stop words are also filtered by using English stop word list of NLTK.

- Feature Vector Representation of Pre-processed data

The next step is concerned with converting the pre-processed data into feature vector .Here the pre -processed data is tokenized into words and converted to numerical representation so that it can be given as input to the machine learning classifier. In our methodology, Word2Vec is used as feature vector which has context related feature vector representation 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.

The Word2Vec feature vector was developed by Tomas Mikolov [11] at Google in 2013.They also enhanced their work by introducing N-gram phrases with Word2Vec [12]. Word2Vec is a shallow neural network feature vector representation technique that produces word embedding which captures semantic relationship between the words. 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.

Word2Vec converts the data corpus to a vector space of several hundred dimensions. The feature vector provides a unique vector to each word on the basis of other context words in the neighborhood of the target word. The word vectors produced are located in the vector space such that the words with similar context are close to each other in the vector space So we can find words with similar context as well as with dissimilar context for the target word. It calculates the cosine similarity distance between the words to find similar/dissimilar 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.

Word2Vec can be used in two ways. They are CBOW (Continuous -Bag Of- Words) and Skip-gram. In CBOW, the target word is predicted using the surrounding context words, Skip-gram uses the opposite technique as compared to CBOW, the surrounding context words are predicted from the target word.

In our methodology, feature vectors are used with Unigrams and Unigrams + Bigrams (with phrases).Unigram refers to single word whereas Bigram refer to sequence of two words. Both CBOW and Skip-gram models of Word2vec are evaluated .N-grams (with phrases) means all the unigrams and bigrams formed are not considered since many of them are not really useful. So those phrases (Unigrams + Bigrams) are taken into consideration which satisfies a threshold value. Here in our experiment we have taken threshold value equal to one and minimum word count also equal to one which refers to frequency of the words considered.

The feature vectors are implemented by using Gensim library[13] of python. Some hyper parameters need to set to obtain the word vector. The following hyper parameters are used for Word2vec for obtaining the feature vector

a) Number of features = 300
b) Minimum word count = 10
c) Number of workers = 4
d) Window Size = 10
e) Down sampling = 1e-3

Number of features refers to the dimension size of the word embedding. Minimum word count refers to a number that words for vector conversion will not be lower than this frequency. Number of workers refers to how many number of threads can be used to train the model. Window size refers to the size of context window to be considered for the target word. Down sampling refers to the threshold value for down sampling the words with higher frequency.

- Sentiment Classification

Finally, Machine Learning based Sentiment Classification is done in our proposed method. The training and testing set of word vectors obtained from the previous step are used in this step. The machine learning based classifiers are trained on the training dataset. After that the testing dataset are used for evaluating the performance of trained classifiers. The classifiers used are Logistic Regression CV,MLP(Multi Layer Perceptron), Random Forest, Decision Tree and Gaussian Naïve Bayes[14][15]. Scikit learn package of python are used for implementing the classifiers.

IV. RESULTS AND DISCUSSION

We have done experimentation for inspecting two things. First is inspecting the performance of feature vector on sentiment classification by using the two models of Word2vec i.e. CBOW and Skip-gram method with their Unigrams and Unigrams+ Bigrams(with Phrases) on car brand reviews. Secondly inspecting the performance of feature vector on classification by taking different number of reviews i.e. gradually increasing the number of reviews in the dataset. The following numbers of reviews are taken for experiment: 5,889 reviews, 15,665 reviews, 25,427 reviews and 39,138 reviews in the dataset.

We evaluated the performance on the basis of Accuracy Score and F1 Score . Accuracy refers to number of correct predictions obtained to the total number of predictions. It predicts the part of prediction our classification model predicted right. F1 score refers the harmonic mean of precision and recall .It selects a classification model on the basis of balance between precision and recall. Precision is the ratio of True Positive with the sum of True Positive and False Positive. Recall is the ratio of True Positive with the sum of True Positive and False Negative.

$$\text{Accuracy} = (\text{True Positive} + \text{True Negative}) / (\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative})$$

$$\text{F1 score} = 2*(\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

TABLE I. COMPARISON OF ACCURACY AND F1- SCORE OF DIFFERENT CLASSIFIERS WITH DIFFERENT DATASET SIZE

Number of Reviews in the Dataset	Classifier	CBOW(Unigrams)		CBOW(Uni +Bigrams) With Phrases		Skip-gram(Unigrams)		Skip-gram(Uni +Bigrams) With Phrases	
		Accuracy	*F1-Score*	*Accuracy*	*F1-Score*	*Accuracy*	*F1-Score*	*Accuracy*	*F1-Score*
5889	**MLP**	0.810413	0.882332	0.820600	0.890045	0.841539	0.904567	0.808149	0.880927
	LRCV	**0.840407**	**0.903623**	**0.835314**	**0.899482**	**0.843237**	**0.905814**	**0.851726**	**0.911307**
	RF	0.838144	0.902055	0.827957	0.895604	0.821166	0.890809	0.824561	0.892882
	DT	0.874363	0.926926	0.791171	0.871204	0.702320	0.804024	0.765139	0.852993
	GNB	0.774759	0.857959	0.782117	0.863233	0.700623	0.802391	0.640634	0.754542
15665	**MLP**	0.811702	0.883415	0.799574	0.874701	0.792766	0.869751	0.802766	0.877072
	LRCV	**0.840426**	**0.903026**	0.830638	0.896543	**0.838511**	**0.901569**	**0.843830**	**0.905315**
	RF	0.816170	0.886970	0.808511	0.882291	0.808085	0.881503	0.800426	0.876449
	DT	0.824468	0.894081	**0.830851**	**0.898816**	0.843830	0.907650	0.796596	0.875683
	GNB	0.749574	0.840190	0.732340	0.827293	0.713404	0.813616	0.651277	0.764816
25427	**MLP**	0.816883	0.887038	0.842837	0.904924	0.809936	0.882075	0.805348	0.878963
	LRCV	**0.844278**	**0.905774**	**0.850046**	**0.909651**	**0.846245**	**0.906912**	**0.846900**	**0.907434**
	RF	0.812951	0.884947	0.828680	0.895782	0.823306	0.892384	0.823568	0.892698
	DT	0.827369	0.895418	0.821995	0.892716	0.750557	0.840259	0.831826	0.901163
	GNB	0.748460	0.839669	0.759602	0.847877	0.744265	0.836312	0.697732	0.801651
39138	**MLP**	0.835122	0.899751	0.817748	0.887640	0.797990	0.873480	0.829416	0.896083
	LRCV	**0.848493**	**0.908323**	**0.846874**	**0.907281**	**0.854199**	**0.911979**	**0.849259**	**0.908819**
	RF	0.831289	0.897850	0.812553	0.885394	0.820644	0.890677	0.809658	0.883503
	DT	0.828734	0.897330	0.769460	0.854408	0.818600	0.890680	0.788707	0.869428
	GNB	0.770397	0.855489	0.755067	0.844759	0.745273	0.837472	0.703798	0.806842

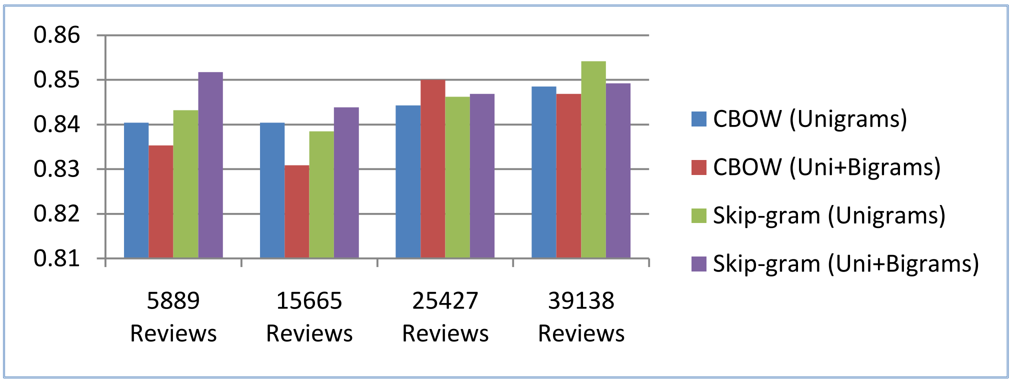

Figure 1 : Comparison Chart showing the Highest Accuracy Score

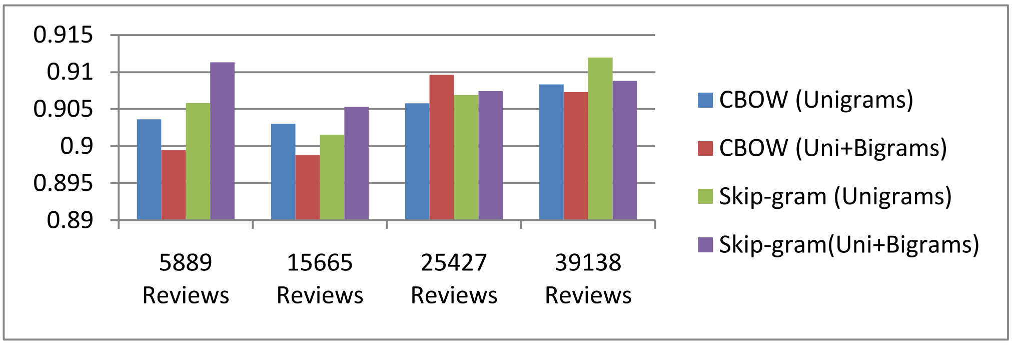

Figure 2 : Comparison Chart showing the Highest F1-Score

The table above shows the results obtained by applying different machine learning classifiers with different dataset size on Word2vec with Unigrams and Unigrams + Bigrams(with Phrases). The highest Accuracy Score and F1- Score obtained with the different dataset size are given in bold text. The Accuracy Score and F1-Score with Logistic Regression CV classifier gives the highest score for all the dataset sizes. The Comparison chart shows the highest accuracy and F1-score for both CBOW and Skip-gram Models. It shows that the best accuracy score and F1 score are obtained with Skip-gram (Unigram) model. It can be seen from the results that as the number of reviews increases in the dataset the feature vector performed well in some cases. The highest scores both in terms of Accuracy and F1-Score are obtained with highest number of reviews i.e. 39,198 reviews. In future, the feature vector can be experimented with other product domain reviews and other feature vectors like Doc2Vec, Glove and Fast Text with other machine learning classifiers or their hybrid can be used to evaluate their performance.

REFERENCES

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

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

[3] B Pang B.,Lee , and L.,Vaithyanathan ”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, 2002.

[4] Barkha Bansal,Sangeet Shrivastava ”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, 2018.

[5] Marwa Naili, Anja Habacha Chaibi, Henda Hajjami Ben Ghezala.”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, 2017.

[6] JoshuaAcosta,NorissaLamaute,MingXio Luo,Ezra Finkelstien,Andreea Cotoranu”Sentiment Analysis of Twitter Messages using Word2Vec,”Proceedings of Student-Faculty Research Day, CSIS, Pace University, Pleasantville ,New York,2017

[7] EissaM.Alshari,AzreenAzman,ShyamalaDoraiswamy,Norwati Mustapha,MustafaAlkheshr.”Improvement of Sentiment Analysis based on Clustering of Word2Vec Features,”28th International Workshop on Database and Expert Systems Applications, IEEE doi 10.1109/dexa.2017.41, 2017.

[8] Sadam Al-Azani, and El-Sayed M. El-Alfy ”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,