Sentiment Analysis of Bengali Comments With Word2Vec and Sentiment Information of Words

Md. Al- Amin
Department of CSE
Shahjalal University of Science and Technology
Sylhet, Bangladesh
alaminbbsc@gmail.com

Md. Saiful Islam
Department of CSE
Shahjalal University of Science and Technology
Sylhet, Bangladesh
saif.acm@gmail.com

Shapan Das Uzzal
Department of CSE
Shahjalal University of Science and Technology
Sylhet, Bangladesh
shoponsustcse11@gmail.com

***Abstract*— The vector representation of Bengali words using word2vec model (Mikolov et al. (2013)) plays an important role in Bengali sentiment classification. It is observed that the words that are from same context stay closer in the vector space of word2vec model and they are more similar than other words. In this article, a new approach of sentiment classification of Bengali comments with word2vec and Sentiment extraction of words are presented. Combining the results of word2vec word co-occurrence score with the sentiment polarity score of the words, the accuracy obtained is 75.5%.**

Keywords— Word Embedding; Word2Vec; Sentiment Polarity; Valence Shifter; CBOW; Skip-gram.

I. INTRODUCTION

In this era of technology, the data are being stored in the internet are growing day by day and the amount of data stored to date is enormous. The trend and behavior of this huge amount of data from different types of people now cannot be determined or analyzed by hand. But these data carry very important information about the opinions of different types of people all over the world, so it has become very necessary to summarize these huge amount of data with some automated systems.

The analysis of polarity of the comments of the people or some specific group of people is called sentiment analysis. Bengali is a highly spoken language and it's spoken by around 189 million people[1]. The analysis of this huge population's opinions has become very challenging and need to introduce new techniques. Many works have been done on sentiment analysis of English language and many of these works has brought significant successes. But, a very few works have been done on sentiment analysis of Bengali language so there are lots of research scopes in this field.

In this article, it is presented that the vector representation of words and the sentiment information of words can be used collectively in the sentiment analysis of Bengali comments. A corpus of single line and multiline comments from Bengali microblogging websites is constructed and each of the comments are tagged to either positive or negative depending on the people's opinions. The people's opinions are taken by creating surveys. Observing that the sentiment classification not only depends on the contextual information of the comments but also depends on the sentiment information of the comments. A new technique of combining these two information is introduced and obtained significant improvements.

II. RELATED WORK

Many research works are done on Sentiment Analysis of English. Cui et al. [2] worked on online product reviews. They classified the reviews to two major classes: positive and negative. They considered around 100k product reviews from different websites. Jagtap et al. [3] applied Support Vector Machine (SVM) and Hidden Markov Model (HMM). Their hybrid classification model to extract the sentiment of teacher feedback assessment performed well. Alm et al. [4] Separated seven emotional words to three polar classes of positive emotional, negative emotional and neutral. They used Winnow parameter tuning approach and got 63% accuracy. Agarwal et al. [5] applied unigram, tree model and feature based model to extract twitter sentiment. Unigram model is outperformed by tree model and feature based model. The accuracy they got is around 61%. Zou et al. [6] introduced a model of learning bilingual word embeddings from a large and unlabeled dataset. They showed that their model outperforms baselines in semantic similarity of words. Turian et al. [7] worked on Brown clusters, embeddings of Collobert and Weston (2008) and hierarchical log-bilinear embeddings. Chen et al. [8] proposed some approaches that can differentiate the released word embeddings models. They showed that embeddings can detect surprising semantics of the sentences even without having the structure. Tang et al. [9] introduced a technique of gathering both contextual and sentiment information of the words by learning Sentiment-Specific Word Embedding. They applied their model to extract twitter sentiment. The accuracy they got is around 83%. Levy et al. [10] Worked on skip-gram model with negative sampling of Mikolov et al.(2013) and generalized it. They extracted dependency based contexts and showed that they produce different types of similarities. Andreas et al. [11] showed three possible benefits of word embeddings: Vocabulary expansion, Statistic sharing and embedding structure. Lebret et al. [12] worked on Word Embeddings with Hellinger PCA. They constructed word co-occurrence matrix to find the contextual representation of words. They obtained around 89% accuracy. Levy et al. [13] applied word embedding models with neural network to determine word similarity and

978-1-5090-5627-9/17/$31.00 ©2017 IEEE.

detect analogy. They achieved better results than the traditional count based distributional models.

A few research works are done on Bengali. Chowdhury et al. [14] Applied Support Vector Machine (SVM) and Maximum Entropy (MaxEnt) to detect the sentiment of Bengali microblog posts. They experimented these two methods by combining them with different types of features. Hasan et al. [15] described a technique of detecting the sentiments of Bengali texts by Contextual Valency Analysis. They applied POS tagger to find total positivity, total negativity and total neutrality and the final results are calculated from these. Das [16] presented a computational approach of tracking the emotions of English and Bengali texts. He classified the emotions into six classes. They are happy, sad, anger, disgust, fear and surprise. Hasan et al. [17] presented a sentiment analyzer that detects the sentiments of Bengali text about a specific subject. Sentiment information is obtained by phrase patterns of positive and negative sentiments and the sentiment orientations. Islam et al. [18] applied Naïve Bayes Model to detect the sentiment of Bengali Facebook Status. By applying Naïve Bayes model along with Bigram, they achieved 0.72 F-score. We [19] worked on word embedding with Hellinger PCA to detect the sentiment of Bengali comments. Word co-occurrence matrix is constructed with skip gram to determine the contextual information of the comments and sliding windows are created to gather similar words in the windows.

III. Methodology

Similar words occur more often in the same context. Word2vec [20] represents each word with a vector representation. In the vector space of word2vec model, the similar words stay closer with each other. Word2vec preserves the syntactic meanings of the words and arranges the words by their syntactic similarity. Thus, in word2vec vector space, similar words by their syntactic structure stay closer but the words of opposite sentiment polarity may also stay closer which brings disastrous results in sentiment classification. So, polarity score of each word has a vital role in sentiment analysis. To overcome this characteristics of word2vec we have introduced a new approach of combining the similarity score of co-occurring words by word2vec and the sentiment polarity score of each word of the query comment.

It's observed that the size of dataset and the accuracy of classification have linear relation, if the dataset size increases then the accuracy also increases. It's a very significant characteristic of word embedding with word2vec.

The comments of the training datasets contain different types of noises. Before we proceed to the training, these noises are eliminated by removing all unwanted punctuation marks, extra spaces, unrecognized characters etc. from the datasets.

Firstly, we have trained 90% of our collected data chosen randomly with word2vec. The dataset are divided into two sets. One set contains all the comments with positive tag and the other one contains all the comments with negative tags. As noise may occur in the corpus, the words which occur at least 10 times in the corpus are taken into the considerations and the other words are ignored as they are less important and have high chances to be as noise or misspelled words. The window size of the training algorithm is assigned 50, it means that each word in the vector space has 50 dimensions of vector. The higher window size provides better results but need larger dataset. The epoch is assigned 15 to iterate the training 15 times.

The positive score and the negative score of a query comment are calculated as follows:

$$P_{\text{syntactic}} = \sum_{i=2}^{m} VecSimilarity_P(W[i-1], W[i]) \quad (1)$$

$$N_{\text{syntactic}} = \sum_{i=2}^{m} VecSimilarity_N(W[i-1], W[i]) \quad (2)$$

Here,

$P_{\text{syntactic}}$ = The syntactic score obtained from the word2vec model by taking the positive corpus as training data set.

$N_{\text{syntactic}}$ = The syntactic score obtained from the word2vec model by taking the negative corpus as training data set.

$W[i]$ = i'th word in the query sentence.

$VecSimilarity_P(x, y)$ = mutual similarity score (ranged from -1.0 to +1.0) of word x and y obtained from word2vec model trained with positive dataset which determines how similar the words are in the vector space (-1.0 is the least similarity score and +1.0 is the best similarity score).

$VecSimilarity_N(x, y)$ = mutual similarity score (ranged from -1.0 to +1.0) of word x and y obtained from word2vec model trained with negative dataset which determines how similar the words are in the vector space (-1.0 is the least similarity score and +1.0 is the best similarity score).

The accuracy we got at this point is not good enough compared to the accuracy of the existing methods of sentiment analysis. We have observed that, there are two major reasons are responsible behind this. They are: (1) The dataset volume and (2) The Sentiment properties of words.

To overcome the first obstacle, we just need to increase the amount of sentences in the dataset. Now, we are collecting more comments to enrich the dataset and the first obstacle can be overcome by this.

The second obstacle can be overcome by taking the sentiment properties of words into consideration. In fact, in sentiment analysis, the most important thing is to determine the sentiment of individual words prior to compute the sentiment of the whole sentence.

Word Embedding with word2vec determines the syntactic properties of words and assigns a score to each of the words in the vector space. The words that occur in the same context are considered more similar than the words that occur in the different contexts. But some words occurring in the same context may have different sentiment polarity. Such as "He is a nice guy" and "He is a poor guy" contain words 'nice' and 'poor' and both words are in the same context, so they are similar with respect to their syntactic structure. But these two words are of opposite polarities. Considering the similarity score of these two sentences thus have a higher probability to be in the same polarity, which is incorrect. To overcome this,

we have brought a new technique and got significant improvements.

We constructed a list that contains highly positive and highly negative Bengali words, especially adjectives. Every word has a polarity score based on their positivity and negativity (ranged from -1 to +1). These scores are calculated by considering the frequencies of the words in the positive and the negative comments. After calculating the syntactic score of the query sentence, the sentiment score of the sentence is calculated as follows:

$$Q_{sentiment} = \sum_{i=1}^{m} Score[W_q[i]] \quad (3)$$

Here,

$Q_{sentiment}$ = Cumulated sentiment score of the query sentence

$W_q[i]$ = i'th word of the query sentence.

$Score[x]$ = sentiment score of word x.

The final scores are calculated by:

$$P_{\text{score}} = P_{\text{syntactic}} + \alpha * Q_{sentiment} \quad (4)$$

$$N_{\text{score}} = N_{\text{syntactic}} - \alpha * Q_{sentiment} \quad (5)$$

Here,

P_{score} = positive score.

N_{score} = negative score.

α = sentiment threshold.

α is corpus dependent and determined by trial and error process until a suitable value is found for which the accuracy is the highest.

Here, the similarity score between each of the co-occurring word pairs of the query comment are taken into considerations. Similarity scores between distant neighbours' words of the comment can also be taken into consideration for higher order contextual information.

The last step is for neutralizing the effects of valence shifter words (না, নি, নাই, নও etc.).

We define another threshold S_{th}, Which is used to detect the highly positive and highly negative words for which we will apply the neutralizing process. For each word x in the query, if (Score[x] >= S_{th} or Score[x] <= - S_{th}) and there exists a valence shifter word y after x, then,

$V_P = \sum_{i=1}^{m-1} Score[W_q[i]]$; For highly positive and valence shifter word pair.

$V_N = \sum_{i=1}^{m-1} Score[W_q[i]]$; For highly negative and valence shifter word pair.

Now, the modified total results will be,

$$P_{\text{score}} = P_{\text{syntactic}} + \alpha * Q_{sentiment} - \beta * (V_P - 2 * V_N) \quad (6)$$

$$N_{\text{score}} = N_{\text{syntactic}} + \alpha * Q_{sentiment} + \beta * (V_P - 2 * V_N) \quad (7)$$

If, $P_{\text{score}} >= N_{\text{score}}$ then the query is positive,

Else the query is negative.

β is another corpus dependent variable like α, which is determined by trial and error process until a suitable value is found for which the accuracy is the highest.

IV. Experimental Setup and Results

A. Sentiment Analysis Dataset

We have collected more than 16,000 Bengali single line and multiline comments from popular blogging websites and tagged each of the comments to either positive or negative by taking opinions from different types of people by surveys. Around 500 people participated in our surveys and we separated our whole dataset to two subsets of positive comments and negative comments by taking their opinions into considerations. As the positive training dataset and the negative training dataset are constructed based on the opinions of various kinds of people, so the clarity of the training datasets is very high and the datasets reflects the actual scenario. Though the actual scenario is reflected by this type of tagging, ambiguity may arise because of the varieties of thoughts of different types of people and it becomes very challenging to tag the comments properly. But, this ambiguity can be reduced by taking a large number of people's opinions into considerations and we have done that.

B. Results and Analysis

We trained our corpus with 90% of the tagged comments chosen randomly and kept the remaining 10% for testing. We trained our model in 6 steps. In each step we trained 2500 new comments from the training dataset. So, our full dataset of 15000 comments is trained by 6 steps. We calculated the accuracy after each step and observed that the accuracy increases with the size of the dataset.

TABLE I. Classification Accuracy

Training Step No.	Data	Accuracy (%)
1	2500 comments	73%
2	5000 comments	74%
3	7500 comments	74.5%
4	10000 comments	74.5%
5	12500 comments	75.5%
6	15000 comments	75.5%

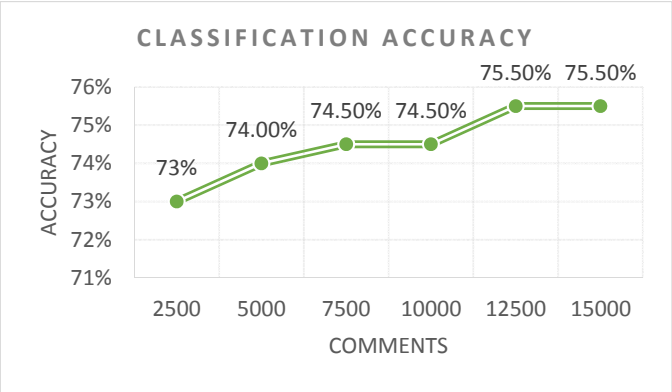

Fig. 1. Classification accuracy.

TABLE II. CONFUSION MATRIX OF TESTING ON 15000 COMMENTS

	Predicted positive	**Predicted negative**
Actually positive	40.5%	12%
Actually negative	12.5%	35%

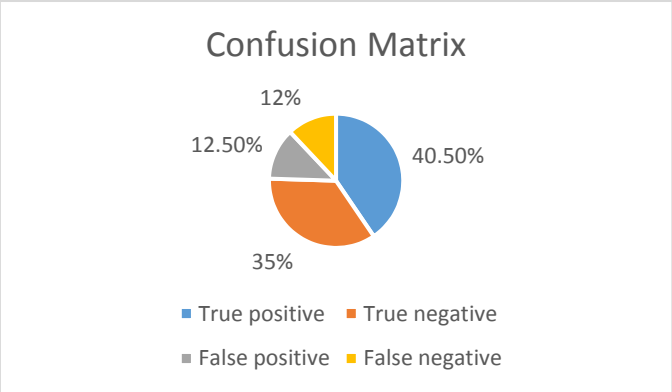

Fig. 2. Confusion Matrix of Testing on 15000 Comments.

V. CONCLUSION

Representation of words within a sentence can define the characteristics of the words. The meanings of the words depend on their contexts. Word embedding of sentences can determine the word characteristics and the context. Other statistical methods applied for Sentiment Analysis of Bengali are largely dependent on the sentence structure. But, By word embedding, the results are independent of the structures of the sentences and the sentiment is defined by the contextual properties of the words. Since Word Embedding is a very new approach in the field of Sentiment Analysis of Bengali Text, it is applied to our own corpus of Bengali Comments collected from recent articles and blogs. A list containing highly positive and highly negative words with their corresponding polarity score is constructed. The accuracy achieved by combining these scores and the valence shifter word neutralization is 75.5% which is very promising and significant for further tuning and research. As it is observed from the results that the accuracy obtained by our model increases with the size of dataset, we are very hopeful that, if a gold standard dataset can be built, then this approach will outperform all the current approaches as shown in the graph. The accuracy graph shows that the accuracy increases with the size of the dataset, so we are working on it to improve the results.

REFERENCES

[1] Lewis, M. Paul, Gary F. Simons, and Charles D. Fennig (eds.), "Ethnologue: Languages of the World," Nineteenth edition. Dallas, Texas: SIL International, 2016.

[2] Hang Cui, Vibhu Mittal and Mayur Datar, "Comparative Experiments on Sentiment Classification for Online Product Reviews," Proceedings of the 21st National Conference on Artificial Intelligence, AAAI, Boston, MA, 2006.

[3] Balaji Jagtap and Virendrakumar Dhotre, "SVM and HMM Based Hybrid Approach of Sentiment Analysis for Teacher Feedback Assessment," International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 3, Issue 3, May-June 2014.

[4] C. Alm and D. Roth and R. Sproat, "Emotions from text: machine learning for text-based emotion prediction," EMNLP, 2005.

[5] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow and Rebecca Passonneau, "Sentiment Analysis of Twitter Data," LSM '11 Proceedings of the Workshop on Languages in Social Media, Pages 30-38, 2011.

[6] Will Y. Zou, Richard Socher, Daniel Cer and Christopher D. Manning, "Bilingual Word Embeddings for Phrase-Based Machine Translation," SemEval, 2012.

[7] Joseph Turian, Lev Ratinov and Yoshua Bengio, "Word representations: A simple and general method for semi-supervised learning," Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 384–394, Uppsala, Sweden, 11-16 July 2010.

[8] Yanqing Chen, Bryan Perozzi, Rami Al-Rfou, and Steven Skiena, "The Expressive Power of Word Embeddings," ICML 2013 Workshop on Deep Learning for Audio, Speech, and Language Processing, Atlanta, USA, June 2013.

[9] Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu and Bing Qin, "Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification," Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 1555–1565, Baltimore, Maryland, USA, June 23-25, 2014.

[10] Omer Levy and Yoav Goldberg, "Dependency-Based Word Embeddings," Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 302–308, Baltimore, Maryland, USA, June 23-25, 2014.

[11] Jacob Andreas and Dan Klein, "How much do word embeddings encode about syntax?," Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), June 2014.

[12] Remi Lebret and Ronan Collobert, "Word Embeddings through Hellinger PCA," Idiap Research Institute, Rue Marconi 19, CP 592, 1920 Martigny, Switzerland, arXiv preprintarXiv:1312.5542, 2013.

[13] Omer Levy, Yoav Goldberg and Ido Dagan, "Improving Distributional Similarity with Lessons Learned from Word Embeddings," Transactions of the Association for Computational Linguistics, vol. 3, pp. 211–225, 2015. Action Editor: Patrick Pantel. Submission batch: 1/2015, Revision batch 3/2015, Published 5/2015.

[14] Shaika Chowdhury and Wasifa Chowdhury, "Sentiment Analysis for Bengali Microblog Posts," International Conference on Informatics, Electronics & Vision (ICIEV), 2014.

[15] K. M. Azharul Hasan, Mosiur Rahman and Badiuzzaman, "Sentiment Detection from Bengali Text using Contextual Valency Analysis," 17th Int'l Conf. on Computer and Information Technology, Daffodil International University, Dhaka, Bangladesh, 22-23 December 2014.

[16] Dipankar Das, "Analysis and Tracking of Emotions in English and Bengali Texts: A Computational Approach," Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, March 28–April 1, 2011.

[17] K. M. Azharul Hasan, Sajidul Islam, Mashrur-E-Elahi and Mohammad Navid Izhar, "Sentiment Recognition from Bangla Text," Technical Challenges and Design Issues in Bangla Language Processing, IGI Global, 2013.

[18] Md. Saiful Islam, Md. Afjal Hossain, Md. Ashiqul Islam and Jagoth Jyoti Dey, "Supervised Approach of sentimentality extraction from Bengali Facebook Status," The 19th International Conference on Computer and Information Technology(ICCIT-2016), December 18-20, North South University, Dhaka, 2016.

[19] Md. Saiful Islam, Md. Al- Amin and Shapan Das Uzzal, "Word Embedding with Hellinger PCA to Detect the Sentiment of Bengali Text," The 19th International Conference on Computer and Information Technology(ICCIT-2016), December 18-20, North South University, Dhaka, 2016.

[20] Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean, "Efficient Estimation of Word Representations in Vector Space," in Proceedings of International Conference on Learning Representations, ser. ICLR '13, 2013.