Exploring Word Embedding For Bangla Sentiment Analysis

Conference Paper · September 2018

DOI: 10.1109/ICBSLP.2018.8554443

4 authors:

Sakhawat H Sumit
BJIT Ltd
5 PUBLICATIONS **26** CITATIONS
SEE PROFILE

Md. Zakir Hossan
Independent University, Bangladesh
4 PUBLICATIONS **10** CITATIONS
SEE PROFILE

Tareq Al Muntasir
Socian Ltd
3 PUBLICATIONS **15** CITATIONS
SEE PROFILE

Tanvir Sourov
Socian Ltd.
2 PUBLICATIONS **11** CITATIONS
SEE PROFILE

Exploring Word Embedding For Bangla Sentiment Analysis

Sakhawat Hosain Sumit, Md. Zakir Hossan, Tareq Al Muntasir and Tanvir Sourov
Socian Ltd
Dhaka, Bangladesh
{sumit, zakir, tareq, tanvir}@socian.ai

***Abstract*—Sentiment Analysis (SA), sometimes known as opinion mining, polarity analysis or emotional AI, is a study of analyzing user's reviews, ratings, recommendations and other forms of online expressions. Most of the research work on SA in Natural Language Processing (NLP) are focused on the English language. However, Bengali is spoken as the first language by almost 230 million people worldwide, 163.9 million of whom are Bangladeshi. These people are found to get increasingly involved in online activities on popular microblogging and social networking sites, sharing opinions and thoughts and most of them are in Bengali and Romanized Bengali (English character to write Bengali) language. These online opinions are changing the way of doing business. And lots of data are being generated each year which are being underutilized. In this paper, we have experimented current state of the art word embedding methods Word2vec Skip-Gram and Continuous Bag of Words with an addition Word to Index model for SA in Bangla language. Word2vec Skip-Gram model outperformed other models and achieved 83.79% accuracy.**

Keywords—Sentiment Analysis, Word Embedding, Bengali Language

I. INTRODUCTION

The rise of social media such as blogs and social networks has reinforced the interest in sentiment analysis. The rapid growth of user's reviews, ratings, recommendations and other forms of online expressions and online opinions have turned into a kind of virtual currency in business industries. It is also a demand for different NGOs, Governments and other organizations. By analyzing these users' sentiments, companies market their products, look for new opportunities and manage their reputation.

As there are an enormous amount of data being generated in every instant, the need for an automated process to filter out the noise and extract relevant insights from these data is increasing rapidly. And thus the increasing demand in the field of sentiment analysis.

The problem is that most SA algorithms use simple terms to express sentiment about a product or service. However, cultural factors, variations, linguistic nuances, word sequences and differing contexts make it extremely difficult to turn a string of written text into a simple meaningful sentiment. Bengali is spoken by almost 300 million people worldwide, 163.9 million of whom are Bangladeshi [1]. A recent study shows that the capital of Bangladesh, Dhaka, is ranked second, considering the number of active Facebook users which is 1.1% of total active users. In Bangladesh 63.3 million of the population out of 163.9 million use the internet and 26.0 million are active social media users [2] and the number is increasing tremendously. These enormous number of users cause rapid growth of reviews, conversations, ratings, recommendations and other forms of online expression and most of them are in Bengali, Banglish or Romanized Bengali language.

In NLP it's a challenging task to capture syntactic and semantic relationships of words from large corpora. Word Embedding is currently the state of the art procedure to overcome this barrier where the words are represented as vectors of continuous real numbers. In recent research, it shows its great self-capability to learn linguistic nuances and features of words from any large text corpora. The idea of Word Embedding originally introduced by Bengio et al [3]. Later in 2013, Mikolov et al. introduced Word2vec model in [4] to learn vector representation of words from a large text corpus, called 'word embedding'. It comes in two flavors, the Continuous Bag-of-Words (CBOW) and the Skip-Gram model. These models are considered cutting edge in word embedding. Currently, Word2vec models are state of the art for language data and very useful for classifying text.

In this paper, we have implemented three different models and two of them are using Word2vec models and another is the traditional Word to Index base text classifier model. We used the vector representation of words generated from skip-gram and CBOW to feed into Deep Long Short-Term Memory (LSTM) network.

II. RELATED WORK

Several recent studies using Artificial Neural Network (ANN) along with Word Embedding have shown promising results in the field of Sentiment Analysis. In [5], SWESA (Supervised Word Embedding for Sentiment Analysis) was introduced which is an algorithm for Sentiment Analysis via Word Embedding. It leverages document label information to learn vector representations of words from a text corpus by minimizing a cost function with respect to both word embedding as well as classification accuracy. A deep learning system for SA of tweets was described in [6]. The main contribution of this work is a new model for initializing the parameter weights of the Convolutional Neural Network (CNN). Authors used an unsupervised neural language model to train initial Word Embedding that is further tuned by deep learning model on a distant supervised corpus. At a final stage, the pre-trained parameters of the network are used to initialize the model. In

978-1-5386-8207-4/18/$31.00 ©2018 IEEE

[7], a model was presented based on Recurrent Neural Network (RNN) and CNN that incorporates the preceding short texts. The model used word2vec method for vector representation of words. The model achieved state-of-the-art results on three different datasets for dialog act prediction.
A textual dataset on Bengali and Romanized Bengali texts was built in [8] and tested in Deep Long Short-Term Memory (LSTM), using two types of loss functions - binary crossentropy and categorical crossentropy. They documented the results along with some analysis of them, which were promising.

A simple and efficient Neural Language Model approach for SA that relies only on unsupervised word representation inputs was proposed in [9]. The model employs Recurrent Neural Network using Long Short-Term Memory (RNN-LSTM), on top of pre-trained word vectors for sentence-level classification tasks. The experiment shows that using word vectors obtained from an unsupervised neural language model as an extra feature with RNN-LSTM for NLP system can increase the performance of the system. Also, simple RNN-LSTM with word2vec achieves an excellent result on IMDB Stanford benchmark for SA task. In [10], a general class of discriminative models was proposed based on recurrent neural network (RNN) and word embedding that can be successfully applied without any task-specific feature and engineering effort. The experimental results on the task of opinion target identification show that RNN, without using any hand-crafted features, outperformed feature rich CRF-based models.

III. Methodology And Approach

Deep learning is part of a broader family of machine learning methods inspired by the structure and function of the brain called ANN. The most important difference between deep learning and traditional machine learning is its performance increases as the scale of data increases. When the data is small, deep learning algorithms do not perform very well. The reason behind that is, deep learning algorithms need a large amount of data to perform well. On the other hand, traditional machine learning algorithms, with their handcrafted rules, fail in this scenario.

A. Long Short-Term Memory

Long Short-Term Memory [11] networks are a type of Recurrent Neural Network (RNN) capable of learning order dependencies in sequence prediction problems. Traditional RNN works well in short-term dependency where RNN is able to learn the ways to connect the recent past information. But it suffers to connect information for long-term dependency problems where the gap between the relevant information is quite large [12]. LSTM networks are capable of learning long-term dependencies. LSTM contains information outside the normal flow of the recurrent network in a gated cell. Information can be stored in, written to, or read from a cell. The cell makes decisions about what to store, and when to allow reading, writing and erasures storing. Gates are used for this purpose based on their strength and importance, which they filter with their own sets of weights. These weights are adjusted via the recurrent network's learning process. The cells learn when to allow data to be entered, left or deleted through the iterative process of making guesses, back-propagating error, and adjusting weights via gradient descent. However, these gates are non-linear, implemented with element-wise multiplication by sigmoids activation and squeeze the data in the range of 0-1. Non-linear gate has the advantage over linear gate of being differentiable, and therefore suitable for back-propagation.

B. Data Wrangling

a) Bangla Web Crawl: We have setup an Apache Nutch crawler [13] which is continuously crawling hundreds of popular Bangla websites. Currently, a large portion of data is coming from various Bangla news portals (eg. Daily Prothom Alo [14], Daily Kaler Kantho [15], Daily Ittefaq [16], BDNews24.com [17] etc) and various Bangla blog sites (eg, Shachalayatan [18], Muktomona [19], Somewhereinblog [20] etc.) Our Apache Nutch setup uses MongoDB [21] as GORA [22] backend and indexes data in Elasticsearch [23]. So far, we have collected 50 Gigabytes of Bangla page crawls and our crawler is continuously crawling new pages.

b) Bangla Sentiment Dataset: To prepare the dataset, we have collected following types of posts from Facebook. Posts from popular Bangladeshi Facebook pages (eg. ProthomAlo [24], Grameenphone [25], RobiFanz [26] etc). Posts from popular Bangladeshi Facebook page followers posted in these pages. Usually, these posts are mostly reviews and customer queries. Posts in popular Bangladeshi groups (eg. Desperately Seeking Dhaka [27]) posted by group members. We wrote a Python script which collects all posts from a given Facebook page ID or group ID and a time range to crawl the posts from. It connects to Facebook Graph API using HTTPS and saves collected posts in a database or CSV file. After collecting the posts, we removed any non-Bengali character and symbols. Then, we split the posts into sentences. We labeled thirty thousand users' positive and negative sentiments where positive and negative sentiments are equally fifteen thousand each. We used the labeled data for SA and for building the word embedding, we used both sentiments and Bangla Web Crawl corpus dataset. Word embedding was used at the top level of our model. Table I presents statistics of the datasets we used.

TABLE I. Datasets Statistics

Datasets	Sentences	Words	Unique Words
Sentiment	1,899,094	23,506,262	394,297
Corpus	24,991,544	623,510,478	1,764,807

C. Experiment

a) Word2vec Word Embedding: We have implemented single hidden layer based two Neural Networks for Skip-Gram and Continuous Bag of Words (CBOW) model for Word2vec Word Embedding using our all sentiments data and Bangla Web Crawl corpus dataset with a vocabulary size of 1,245,974, embedding vector length of 150 and window size of 5. We included words in our vocabulary which appears at least two times in our whole data corpus. Fig. 1 and 2 present Word2vec Skip-Gram and CBOW word

embedding result of Bengali months, number and similar words of Bangladesh (বাংলাদেশ).

b) Training: We implemented a Neural Network using a single LSTM layer with Rectified Linear Unit [28] activation function and a two-unit based fully-connected layer and experimented with grid search using epoch, batch size and number of neurons in the layer as hyper-parameters. All the weight matrices were initialized with the normalized initialization [29] and the bias vectors were initialized to 0. We kept the learning rate at 0.001 and dropout regularization [30] rate at 0.70 constantly. Dropout regularization was used for each network to avoid overfitting problem. We used ADAM as the optimization method [31]. The softmax output and the categorical cross entropy cost were combined as the model cost.

Table II shows the grid search's hyper-parameters. We performed three different grid-searches using the same hyper parameters for skip-gram, CBOW, and word to index-based models. Figure 3, 4 and 5 present the grid search results respectively in skip-gram, CBOW, and word to index. Fig. 6 to 8 shows the model's accuracy graph for training and validation respectively in skip-gram, CBOW, and Word to Index based models.

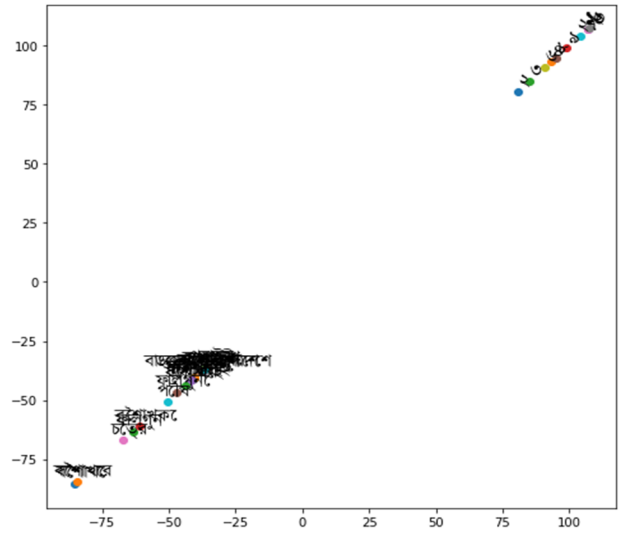

Fig. 1. Word2vec Skip-gram Word Embedding Result

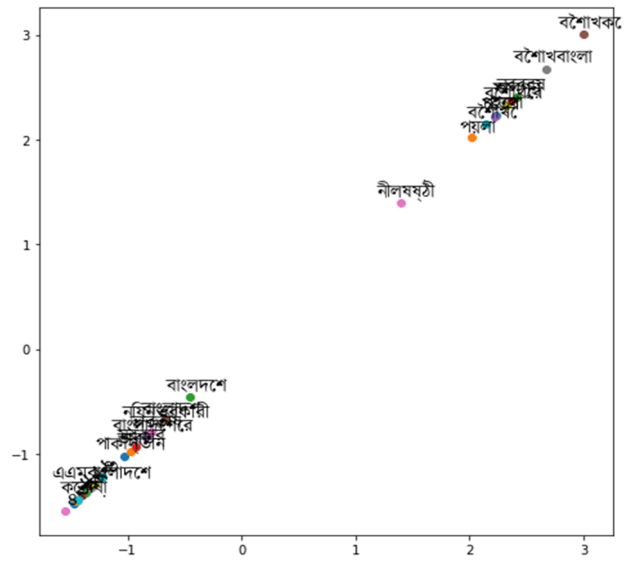

Fig. 2. Word2vec Continuous Bag of Words Word Embedding Result

TABLE II. HYPER-PARAMETERS

Epochs	4	6	8	10
Batch Size	64	128	256	
Neurons	8	10	12	

TABLE III. OPTIMAL HYPER-PARAMETERS AND ACCURACY FOR MODELS

Models	**Epochs**	**Neurons**	**Batch Size**	**Accuracy**
Skip-Gram	10	10	128	**83.79%**
CBOW	10	10	256	82.57%
Word to Index	10	10	128	54.40%

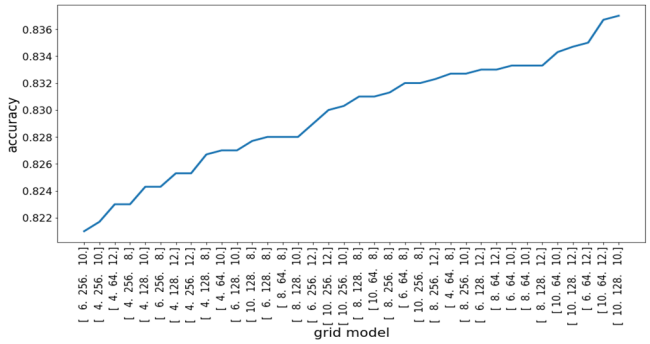

Fig. 3. Accuracy vs Model Parameters for Skip-Gram Base Grid Search

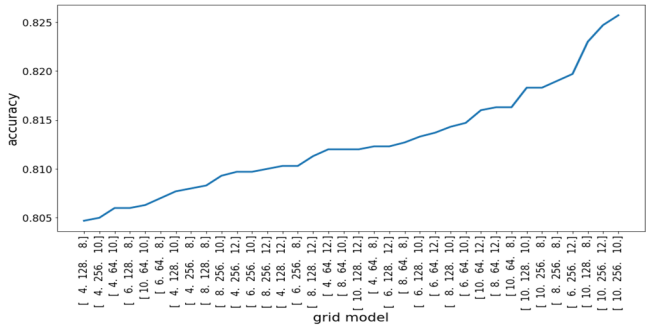

Fig. 4. Accuracy vs Model Parameters for CBOW Base Grid Search

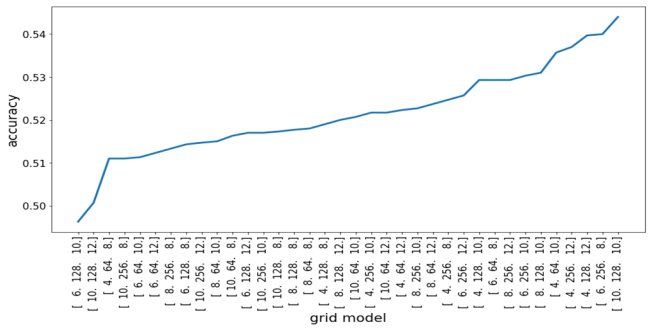

Fig. 5. Fig 5: Accuracy vs Model Parameters for Word to Index Base Grid Search

IV. Result And Discussion

Word2vec skip-gram and CBOW models are almost identical in their algorithmic construction. The core difference between these models is skip-gram predicts source context-words from the target words, while CBOW does the inverse and predicts target words from source context words. From statistics, CBOW has a smoothness property which delivers a lot of distributed information for treatment in an entire context as one observation. In contrast, skip-gram treats each context-target pair as a new observation. This leads skip-gram to perform better for large data corpus. From figure 1 and 2, we can observe that skip-gram has better word representations than CBOW for our data corpus.

Table 3 presents optimal hyper-parameters and accuracy for different models from grid search. From our grid search results, we found that Word2vec Word Embedding based models achieved much higher accuracy compare to Word to Index based models and skip-gram based model outperformed other models due to better word representations. Our optimal model achieved accuracy rate 83.79%.

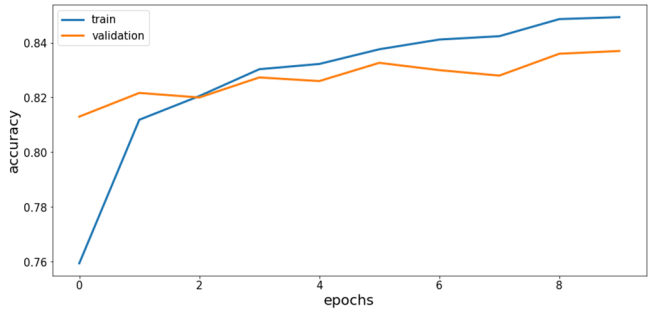

Fig. 6. Skip-Gram Based Optimal Model Accuracy

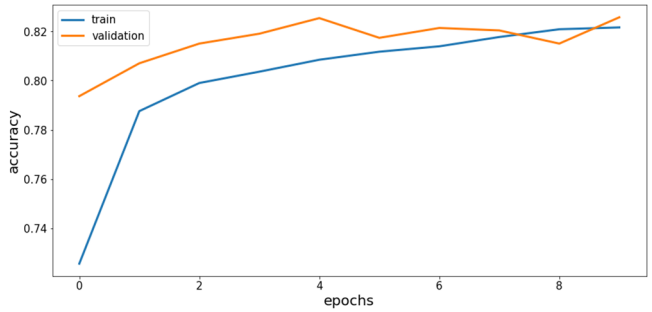

Fig. 7. CBOW Based Optimal Model Accuracy

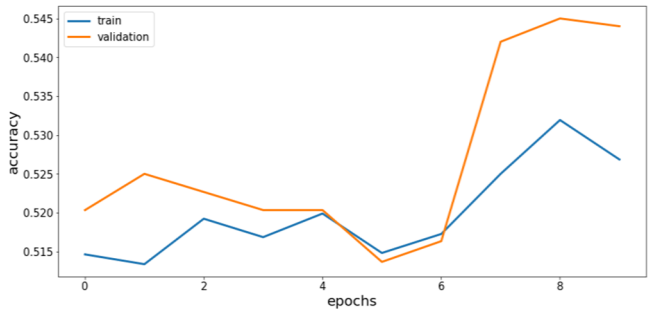

Fig. 8. Word to Index Based Optimal Model Accuracy

V. Conclusion And Future Work

In this work, we have experimented with two-word embedding methods and a traditional word to index-based method for Bangla sentiment analysis. In the very first level of our method, we have build single layer based network for embedding words using word2vec approaches. Then we feed the embedding into the recurrent layer to discriminate sentiments. We found that skip-gram based method outperformed other due to better word representation.

In future, we will publish our dataset as open source and label more sentiment data for different classes (Positive, Negative, Neutral and Question). Even though our model achieved state of the art accuracy rate for Bengali language, it needs more experiment with additional data and different Neural Networks architecture.

References

[1] Wikipedia, Bengalis, https://en.wikipedia.org/wiki/Bengalis, Accessed 26 Sept 2017.

[2] Mhamud Murad , Dhaka ranked second in number of active Facebook users,BDnews24,http://bdnews24.com/bangladesh/2017/04/15/dhakar anked-second-in-number-of-active-facebookusers, Accessed 16 Sept 2017.

[3] Y. Bengio, R. Ducharme, P. Vincent and C. Jauvin, "A neural probabilistic language model", Journal of machine learning research, 2003, vol. 3, pp.1137-1155.

[4] T. Mikolov, S. Ilya, C. Kai, S.C. Greg and D. Jeff, "Distributed representations of words and phrases and their compositionality", In Advances in neural information processing systems, 2013, pp. 3111-3119.

[5] P.K Sarma, and B. Sethares, " Sentiment Analysis by Joint Learning of Word Embeddings and Classifier", arXiv preprint arXiv:1708.03995, 2017.

[6] A. Severyn and A. Moschitt, "Twitter sentiment analysis with deep convolutional neural networks", InProceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2015, pp. 959-962.

[7] J.Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks", arXiv preprint arXiv:1603.03827, 2016.

[8] A. Hassan, M.R. Amin, N. Mohammed and A.K.A. Azad, "Sentiment Analysis on Bangla and Romanized Bangla Text (BRBT) using Deep Recurrent models", arXiv preprint arXiv:1610.00369, 2016

[9] A. Hassan, "Sentiment Analysis With Recurrent Neural Network And Unsupervised Neural Language Model", IEEE Signal Processing Society, 2017.

[10] P. Liu, S. Joty and H. Meng, "Fine-grained opinion mining with recurrent neural networks and word embeddings", In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1433-1443.

[11] S. Hochreiter and J. Schmidhuber, "Long short-term memory", Neural computation, 1997, vol. 9(8), pp.1735-1780.

[12] Y. Bengio, P. Simard and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult", IEEE transactions on neural networks, 1994, vol. 5(2), pp.157-166.

[13] Apache Nutch, http://nutch.apache.org/, Accessed 20 Jun 2017.

[14] Prothom Alo, http://www.prothomalo.com/, Accessed 10 Dec 2017.

[15] Kalerkantho, http://www.kalerkantho.com/, Accessed 10 Dec 2017.

[16] Ittefaq, http://www.ittefaq.com.bd/, Accessed 10 Dec 2017.

[17] Bdnews24, https://bdnews24.com/, Accessed 10 Dec 2017.

[18] Sachalayatan , http://www.sachalayatan.com/, Accessed 10 Dec 2017.

[19] Mukto-mona, https://blog.mukto-mona.com/, Accessed 10 Dec 2017.

[20] Somewhereinblog, http://www.somewhereinblog.net/, Accessed 10 Dec 2017.

[21] Mongodb, https://www.mongodb.com/, Accessed 10 Dec 2017.

[22] Gora, http://gora.apache.org/, Accessed 10 Jun 2017.

[23] Elastic Search, https://www.elastic.co/, Accessed 15 Jun 2017.

[24] ProthomAloFacebook, https://www.facebook.com/DailyProthomAlo/, Accessed 15 Jun 2017.

[25] GrameenphoneFacebook, https://www.facebook.com/Grameenphone/, Accessed 15 Jun 2017.

[26] RobiFanz Facebook, https://www.facebook.com/RobiFanz/, Accessed 15 Jun 2017.

[27] Desperately Seeking Dhaka Facebook, https://www.facebook.com/groups/dsdbangladesh/about/, Accessed 15 Jun 2017.

[28] A. Krizhevsky, I. Sutskever and G.E. Hinton, "Imagenet classification with deep convolutional neural networks", In Advances in neural information processing systems, 2012, pp. 1097-1105.

[29] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", In Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010, pp. 249-256.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting", The Journal of Machine Learning Research, 2014, vol. 15(1), pp.1929-1958.

[31] D.P. Kingma and J. Ba, "Adam: A method for stochastic optimization", arXiv preprint arXiv:1412.6980, 2014.