An Approach for Bengali Text Summarization using Word2Vector

Conference Paper · July 2019

DOI: 10.1109/ICCCNT45670.2019.8944536

Sheikh Abujar
Daffodil International University

Abu Kaisar Mohammad Masum
Daffodil International University

Syed Akhter Hossain
Daffodil International University

An Approach for Bengali Text Summarization using Word2Vector

Sheikh Abujar
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
sheikh.cse@diu.edu.bd

Abu Kaisar Mohammad Masum
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
mohammad15-6759@diu.edu.bd

Md Mohibullah
Dept. of CSE
Comilla University
Cumilla,Bangladesh
mohib.cse.bd@gmail.com

Ohidujjaman
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
jaman.cse@diu.edu.bd

Syed Akhter Hossain
Dept. of CSE
Daffodil International University
Dhaka, Bangladesh
aktarhossain@daffodilvarsity.edu.bd

***Abstract*— Text Summarization is one of the mentionable research areas of Natural language processing. Several approaches have already been developed in this concern. Such as – Abstractive approach and extractive approach. Most recent recurrent neural network methods are producing much better results. Several mentionable research has already been discussed for English language summarizer, but a few have already done for the Bengali language. There are so many prerequisites for data analysis purpose -word2vector is one of them. Understanding the vector representation of any text leads the way to identify the key main points of that specific text and helps to measure the relationship of that text with other texts in similarity/dissimilarity [11]. Generated matrix using word2vector can easily applicable for identifying top-ranked sentence/words, either domain specific or in general form. In this paper, a word2vector approach has been discussed in the context of text summarization for the Bengali language.**

Keywords*— *Word2vector, Natural Language Processing, Text Summarization, Bengali text analysis.

I. INTRODUCTION

Word2vec is one kind of neural network that uses two layers to process data. Because of that though it is a neural network, but not a deep neural network, as deep neural network uses more layer than word2vec. In a word2vec it uses text corpus as input data and as outcome it returns set of vector. It can feature vectors for different words in the corpus given to the system. Word2vec turns general input corpus data into numerical form, so that the deep net can understand it and data can be analyzed easily. General parsing is not the only application of word2vec, moreover it extends beyond it. Word2vec can be applied in different things, some of them are social media graphs code, playlists, and other verbal or symbolic series, because in those kind of data patterns can be discerned. Because like other different text data words are also the simple discrete state. For example, the probability of being co-occur. If usefulness and purpose of word2vec is considered, it can be said that it group the vectors of similar words and according to the match it combine them together in vector space. It uses mathematical terms to find out similarities. Without human intervention word2vec can create word vectors. Those vectors are mainly distributed numerical representations, which is similar as word features such as the context of individual words. If enough data is provided to it, word2vec can produce extremely highly accurate output data. In a word2vec neural network the output is a vocabulary. In this vocabulary each item is attached with a vector. Deep neural network uses it to find relationship between words [12].

II. LITERATURE REVIEW

Word2vec is a very large topic to explore and development. All over the people many people are working on this topic to improve the result as well as analysis process. Based on the research on this topic the study can be divided into two parts. One of them is development and another one is application. In this section few research will be discussed shortly.

Wang Ling, et al (2015) worked on Syntax problems for word2vec adaptation [1]. They presented a model that contains two simple modifications in the worldwide popular word2vec tool. They did this in order to generate embedding more suited to tasks that are involving syntax. In their research they proposed a model to improve parts of speech tagging and dependency parsing. Researcher Dongwen Zhang and his team worked on Chinese language comment’s sentiment analysis using word2vec [2]. They also used SMV in their research. In their research they used combined approach of SMV and word2vec in order to extract semantic relationships between words. Joseph Lilleberg et al (2015) worked on SVM and word2vec and their focus was text classification [3]. This work explains several usefulness of word2vec. They have stated tf-idf and word2vec can be used together and the result is much improved. Because alone word2vec provides complementary features that is not more efficient than their model.

Researcher Bai Xue et al (2014) had worked on sentiment computing as well as classification from the data collected from sina weibo using word2vec [4]. According to their research as weibo is used by a good amount of people now a days, therefore they proposed a model to analyze the sentimental state of text data in weibo. Chinese researcher Yao Yao and his team worked on Sensing spatial distribution of land which is developed in combination of points-of-interest with the help of Google Word2Vec model [5]. Satwik Kottur et al (2016) used word2vec for embedding visually grounded word [6] throughout a learning model. In this research a new proposed model has been developed, which helps to learn the visually grounded word embedding's so that it can be used to capture semantic relatedness visual notions. In 2015 Long Ma and Yanqing Zhang was researching on processing of big data using word2vec [7]. In this research they first trained the data with the help of Word2Vec model and after that evaluated similarity of the w

Andi Rexha's team's research topic in 2016 was sentiment classification of tweet data using word2vec [8]. They presented a Word2Vec approach in order to autonomously predict the polarity class emotion of a target phrase of a tweet. On the other hand, Shihao Ji, et al worked on parallelizing word2vec in shared and distributed type of memory [9]. In this research they improved the algorithm of word2vec for parallelizing the word2vec in memory system. Bangladeshi researcher Md. Al-Amin et al. worked on Sentiment analysis and their target data was Bengali comments which was analyzed by Word2Vec technique [10]. They used this techniques to extract sentiment related information from text data. In their research they used word2vec for sentiment classification of Bengali comments using a new model and on the other hand they have extracted the sentiment using word2vec, word co-occurrence score with the sentiment polarity score.

Word embedding is important for text analysis. It carries the numerical value of the related word in a document file. This embedded word file helps to analysis any text documents such as making text summarizer, bi-directional text generation etc. There are several pre-trained word embedded file present different kinds of language but few of word embedded file in our Bengali language. So our main intention in this paper to rich NLP research for our Bengali language and introduce a method for making a word embedding for the Bengali language.

III. Methodology

Word2vec is used to produce word embedding. There are some reasons why we used word embedding such as the concept of a word is not understood by a machine. A machine can understand only binary or numerical value. So, process a language and working with natural language processing word2vector must be needed. When applying word embedding every machine can convert tokenize word to a vector where each vector represents vocabulary of text documents. Word2vec contain 2 layers of a neural network which is not deep. It contains several dimensions with unique word of the text document. Our workflow is given below in figure 1 for making Bengali text word2vec representation.

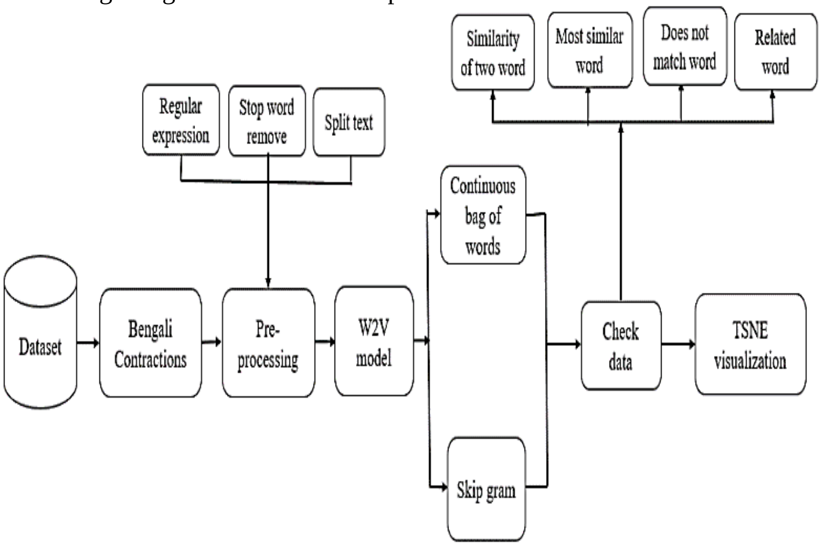

Figure1: Working flow for Bengali Word2vec Representation

A. Data Collection and Preprocessing

Our dataset contains 1k Bengali news article and their summary of each article. We collect data from online news portal and social media pages. For word embedding, we use clean Bengali text and tokenized prepare them for input of the neural network and neural network provide a numerical value for each word. Where every vector represents our dataset vocabulary.

Before embedding word, we need to processing data. Processing of Bengali text data is quite difficult such as remove space from word or sentence, remove unwanted character etc. At first, we need to add Bengali contraction in dataset word because of contraction uses the short form of the word, but we need a full form of the word for embedding.

SHORT FORM	**FULL FORM**
মি.	মিস্টার
রেজি:	রেজিস্ট্রেশন
ডা.	ডাক্তার

Table1: Example of Bengali contractions.

Then we split the text and remove unwanted things such as whitespace, Bengali digits, English character, and punctuation and remove stop words from the text which are unnecessary. Finally, create a clean text with a summary for use as an input of the model

B. Word2vec Model

Word2vec contains a shallowest neural network use to learn the status of word in a text document. Each vector represents a word with a numerical value and provides a semantic description of document words. We used 2 methods for word embedding one is Skip-Gram and another is Continuous Bag of Word Model (CBOW). Continuous Bag of Words model is used word context to predict a target word corresponding to the context. Skip-gram is using a word to predict the value of the target or goal word context.

a. Problem Assertion:

Word2vec are found the similarity and dissimilarity of the words containing in the dataset. In the word vector, it uses the word offset technique which contains general algebraic operation. Let consider, the vector of Bengali words such as the vector 'রাজা' - the vector 'পুরুষ' + the vector 'নারী' now the result will be nearest vector value of 'রানী'. This paper we have tried to discuss about how easy to produce a Bangla word to vector for working text documents, how Bengali word vector represents the similarity, dissimilarity of the words and their relationship with each other in vector context.

b. Skip-Gram Model

This model tries to identify the words based on other words in a similar sentence. As input, we use the current word as an input with a hidden projection layer which predicts the word in the range. Distance words are less related where current words are closely related.

Algorithm1 for genism Skip-gram model

```
1: import model
2: Define Word2Vec(size, window, minimum count, sg, workers, hs, negative)
3: Build vocabulary(input text)
4: Define train(sentences=input text, total examples=length of input text, epochs number)
5: End
```

The formula for the model is,

$$Q = C \times (D + D \times \log_2(v)) \qquad (1)$$

Here, C=Maximum distance of word

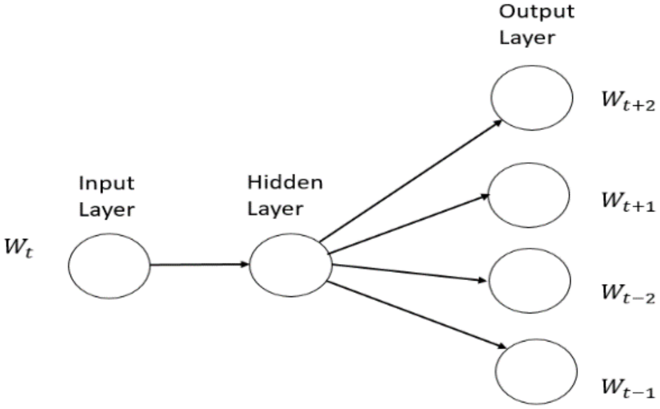

Figure2: View of Skip-Gram Model

c. Continuous Bag of Word Model (CBOW)

Here hidden layer is removed and projection shared all words where all words get the same position. This way is called Bag of word. But its continuously distributed word that's why called Continuous Bag of Words model.

Algorithm2 for genism CBOW model

```
1: import model
2: Define Word2Vec(Input text, window, minimum count, workers)
3: Define train(input text, total examples, epochs number)
5: End
```

The formula for the model is,

$$Q = N \times D + D \times \log_2(v) \qquad (2)$$

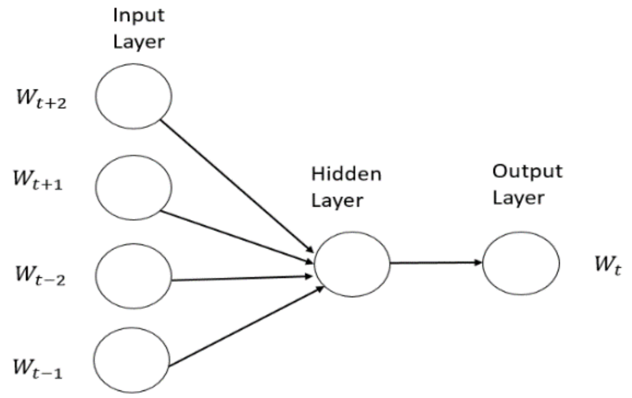

Figure3: View of Continuous Bag of Word Model (CBOW)

IV. EXPERIMENT AND OUTPUT

Working with Bengali text is very challenging every time such as read Bengali dataset, Bengali text processing, remove stop word, regular expressions for Bengali clean text. But after all of those step successfully completes and we apply a clean text for Bengali word embedding. We train those words using Continuous Bag of Word Model (CBOW) and Skip-Gram Model and for visualizing the word we use T-distributed Stochastic Neighbor Embedding (TSNE). Provided some table in below which contains our experiment result and where we show the similarity of the word, the most similarity of the word, does not match between words, get the related term of the word.

Similar word	**Model**	**Value**
"অনুভূতিতে", " অনুষ্ঠানে "	CBOW	0.73497474
	Skip-Gram	0.23969087

Table1: Similarity of two words

Most Similar word	Model	Word and Value
" রয়েছে "	CBOW	'সার্জেন্ট'= 0.922 'মালাইকা'=0.907 'পাচার'= 0.899 'প্রবেশ'=0.889 'ফোন'=0.881
	Skip-Gram	'ডিসেম্বর'= 0.834, 'ডিএসইর'= 0.823 'সাত'= 0.804 'ক্যান্টনমেন্ট'= 0.804 'থেকে'= 0.799 'ডিএসইএক্সের'= 0.796 'বিজিএমইএর'= 0.792 'দিনই',=0.792 'সার্কিট'= 0.788 'জমা'= 0.785

Table2: Most Similar word measure

Does not Similar word	Model	Word
"আদেশ","রয়েছে","হাইকোর্ট"	CBOW	"আদেশ"
	Skip-Gram	"আদেশ"

Table3: Does not similar word measure

Related Term	Word and Value
" চিন্তা "	গঠনের 0.976 দমন 0.974 সেপ্টেম্বর 0.973 দুদক 0.967 ভোলা 0.959 মূলধারায় 0.954 পক্ষে 0.953 শিল্পমন্ত্রী 0.952 অভিযোগের 0.951 জিজ্ঞাসাবাদের 0.951

Table 4: Related term measure

Finally, we visualize the graph of the Bengali words which represent the numerical value of the Bengali text document. When we define tsne set the labels and token of words. In tsne model set the value of perplexity=40, set iteration=2500, init='pca', set n_component=2, and set the figure size=(16,16).For show Bengali word we use Bengali font properties.

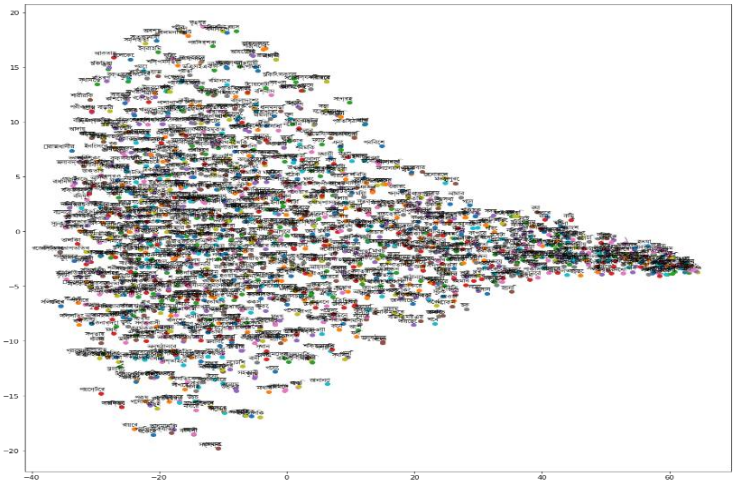

Figure 4: TSNE visualization for Bengali Word2vec

V. Conclusion and Future work

Word embedding is very important when we working with a text document. All vector carries the vocabulary of a text document. It puts all similar word group in vector space and also measures the similarity of the word with each other. In this paper, we embedding Bengali text which was collected from online. Our collected dataset contains Bengali text and their summary. We were able to create a better Bengali word embedding file for applying dataset.

The main limitation of this paper is limited to vocabulary. Because the available dataset is not enough in Bengali Language but our research purpose we collect several data from a different website and social media and embedded them. Another limitation is the sentence structure of Bengali language, it is difficult to accurately divide word from Bengali the sentences.

Word2vec is important when working with text such as text summarization. It keeps all similar word in a numeric value which helps when LSTM cell working with important or non-important value. Here we working with a medium dataset for making word embedding in Bengali text and we try to make a good Bengali Word2vec file. Our applying model gives a better output but in future, we want to make a big word embedding file for Bengali text applying a very big dataset and want to improve our Bengali language research resource in natural language processing.

VI. Acknowledgment

We want to give thanks to our Computer Science and Engineering Department to provide a better facility for research. Specially thanks to our DIU-NLP & ML lab for supporting and helping to complete our research project.

References

[1] W. Ling, C. Dyer, A. Black, I. Trancoso, “Two/Too Simple Adaptations of Word2Vec for Syntax Problems,” in Human Language Technologies: The 2015 Annual

Conference of the North American Chapter of the ACL, pages 1299–1304, Denver, Colorado, June, 2015.

[2] D. Zhang, H. Xu, Z. Su, Y. Xu, “Chinese comments sentiment classification based on word2vec and SVM,” in Expert Systems with Applications, Volume 42, Issue 4, Pages 1857-1863, March 2015.

[3] J. Lilleberg, Y. Zhu, Y. Zhang, “Support vector machines and Word2vec for text classification with semantic features,” in 2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing, DOI: 10.1109/ICCI-CC.2015.7259377, Beijing, China, September 2015.

[4] B. Xue, C. Fu, Z. Shaobin, “A Study on Sentiment Computing and Classification of Sina Weibo with Word2vec,” in 2014 IEEE International Congress on Big Data, DOI: 10.1109/BigData.Congress.2014.59, ISSN: 2379-7703, USA, September 2014.

[5] Y. Yao, X. Li, X. Liu, P. Liu, Z. Liang, J. Zhang & K. Mai, “Sensing spatial distribution of urban land use by integrating points-of-interest and Google Word2Vec model,” in International Journal of Geographical Information Science, 31:4, 825-848, DOI: 10.1080/13658816.2016.1244608, October 2016.

[6] S. Kottur, R. Vedantam, J. M. F. Moura, D. Parikh, “Visual Word2Vec (vis-w2v): Learning Visually Grounded Word Embeddings Using Abstract Scenes,” in IEEE Conference on Computer Vision and Pattern Recognition, pp. 4985-4994, 2016.

[7] L. Ma and Y. Zhang, "Using Word2Vec to process big text data," in 2015 IEEE International Conference on Big Data (Big Data), pp. 2895-2897, DOI: 10.1109/BigData.2015.7364114, Santa Clara, CA, 2015.

[8] A. Rexha, M. Kröll, M. Dragoni, R. Kern, “Polarity Classification for Target Phrases in Tweets: A Word2Vec Approach,” in The Semantic Web. ESWC 2016. Lecture Notes in Computer Science, vol 9989. Springer, Cham, 2016.

[9] S. Ji, N. Satish, S. Li and P. Dubey, "Parallelizing Word2Vec in Shared and Distributed Memory," in IEEE Transactions on Parallel and Distributed Systems, DOI: 10.1109/TPDS.2019.2904058, 2019.

[10] M. Al-Amin, M. S. Islam and S. Das Uzzal, "Sentiment analysis of Bengali comments with Word2Vec and sentiment information of words," in International Conference on Electrical, Computer and Communication Engineering (ECCE), pp. 186-190, DOI: 10.1109/ECACE.2017.7912903, Cox's Bazar, 2017.

[11] Abujar S et al (2017) A heuristic approach of text summarization for Bengali documentation. In: 8th IEEE ICCCNT 2017, IIT Delhi, Delhi, India, 3–5 July 2017

[12] Abujar S, Hasan M (2016) A comprehensive text analysis for Bengali TTS using Unicode. In: 5th IEEE international conference on informatics, electronics and vision (ICIEV), Dhaka, Bangladesh, 13–14 May 2016

[13] Abujar S., Hasan M., Hossain S.A. (2019) Sentence Similarity Estimation for Text Summarization Using Deep Learning. In: Kulkarni A., Satapathy S., Kang T., Kashan A. (eds) Proceedings of the 2nd International Conference on Data Engineering and Communication Technology. Advances in Intelligent Systems and Computing, vol 828. Springer, Singapore