2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), 24-25 December, Dhaka

Bangla Word Prediction and Sentence Completion Using GRU: An Extended Version of RNN on N-gram Language Model

Omor Faruk Rakib, Shahinur Akter, Md Azim Khan, Amit Kumar Das, Khan Mohammad Habibullah

Department of Computer Science and Engineering
East-West University
Dhaka, Bangladesh
Email: rakib1001@gmail.com, shahinur.shoron@gmail.com, azimkhantonmoy@gmail.com, amit.csedu@gmail.com, h.rana10@ewubd.edu

Abstract— **Textual information exchange, by typing the information and send it to the other end, is one of the most prominent mediums of communication throughout the world. People occupy a lot of time sending emails or additional information on social networking sites where typing the whole information is redundant and time-consuming in this advanced era. To make textual information exchange more speedy and easier, word predictive systems are launched which can predict the next most likely word so that people do not have to type the next word but select it from the suggested words. In this study, we have proposed a method that can predict the next most appropriate and suitable word in Bangla language, and also it can suggest the corresponding sentence to contribute to this technology of word prediction systems. This proposed approach is, using GRU (Gated Recurrent Unit) based RNN (Recurrent Neural Network) on n-gram dataset to create such language models that can predict the word(s) from the input sequence provided. We have used a corpus dataset, collected from different sources in Bangla language to run the experiments. Compared to the other methods that have been used such as LSTM (Long Short Term Memory) based RNN on n-gram dataset and Naïve Bayes with Latent Semantic Analysis, our proposed approach gives better performance. It gives an average accuracy of 99.70% for 5-gram model, 99.24% for 4-gram model, 95.84% for Tri-gram model, 78.15%, and 32.17% respectively for Bi-gram and Uni-gram models on average.**

Keywords—word prediction, sentence suggestion, Bangla language, GRU based RNN, n-gram language model

I. INTRODUCTION

To communicate with each other in this modern era, we use different devices to type down the textual information and send it to the other end, but it is redundant to type the whole text again and again, and also it requires time to type the entire text. In that case, the textual word prediction system can ease our life where it will predict the next plausible word, and we could pick those most likely predicted words instead typing those. The problem of guessing which words are expected to follow a sequence of words or a given segment of text is called word prediction [3]. It is an 'intelligent' feature that can alleviate writing breakdowns by merely reducing the number of keystrokes necessary for typing words [5]. The higher the number of keystrokes that the user saves, the better the performance as it reduces both times and attempts for producing a text [5]. In a computer program when the user inputs a word or multiple words, it will present a list of possible words for that particular input, and also when the intended word appears in the list, users can click it, and that will insert the word into the document [2]. Many people in the world are physically, perceptively or cognitively challenged and are slow typists. These types of people can live a comfortable life if they have the scope of typing anything easily being aided through automatic sentence completion technique by the process of word prediction. Also, word prediction can help early learners like students or novice researchers to make fewer spelling errors and enhance the speed of typing. Considering the welfare of word prediction system, several research works have been done where the word prediction system was implemented using different methods to reduce the time required to type and make life easier. Among them, P. Barman and A. Boruah has mentioned in their research work [1] that they have used LSTM (Long Short Term Memory) with RNN (Recurrent Neural Network) to predict the next possible word in Assamese Language and got 88.20 % accuracy for Assamese text and 72.10 % accuracy for phonetically transcript Assamese language. Again, authors of paper [4] proposed a model that can significantly predict most desired words while typing, and their research has experimented on personal emails, call-center emails, weather reports and cooking recipes. They have developed an evaluation metric and adapted N-gram language models to predict subsequent words that require an accurate performance metric rather than the customary performance metric.

We have studied many techniques to predict the next words of a sentence in different languages, especially for the English Language, and there was little satisfactory researches on Bangla language to predict words in a sentence or to predict the whole sentence [6]. Contributing to this research field, a novel approach is proposed in this study to predict the next appropriate word (One or more) and suggest sentences in Bangla language. In our proposed approach we have used GRU based RNN on n-gram dataset to create such language models, which can predict the next most likely Bangla word(s) from a given sequence of input words. As this research work is dealing with sequential data, the next

978-1-7281-6099-3/19/$31.00 ©2019 IEEE

word will be predicted based on not only one word but one or several words and based on their subsequent order; hence RNN is used to train the dataset. Then again, GRU is also used to solve the vanishing gradient problem, a significant drawback of basic RNN. However, the proficiency of this model largely depends on the dataset that we are going to use, so we have tried to collect the dataset from different sources such as the daily "Prothom-Alo" newspaper [7], BBC Bangla news [8], and Bangla academic books [9].

Although there are several research works on word prediction in Bangla language using machine learning approaches, but yet it requires to upgrade the system for better accuracy, and our proposed method has improved the word prediction system providing more accurate and efficient predictions.

The overall contribution of this research work is-

- As per our knowledge, no research work has been done using the same method for Bangla language that we have proposed.
- This method can suggest complete sentences simultaneously with most likely next word prediction for Bangla language.
- A large dataset is used to escalate the accuracy of our proposed method, which wasn't used earlier for Bangla word prediction.
- Our proposed method gives better accuracy than other methods that researchers have used.

The rest of the paper is constructed as follows. Section II will discuss the related works, section III will represent Methodology including dataset overview and implementation, section IV will show the result analysis, and lastly, section V will describe the limitations and future works as the conclusion.

II. Related Works

In recent years, several research works have been done regarding textual word prediction system to find the most suitable and appropriate next word, where Bangla word prediction and completing a sentence is one of these research topics [21] [22] [23] [24] [25] [26]. In the paper [2], authors have considered the disabled people and students who use keyboards for searching from the websites and proposed a model that will predict next several words rather than one word to complete the sentence. For this work, they have used stochastic, i.e. N-gram modeling process. Also, they have used a large Bangla corpus of different type words to find better performance.

Again to improve the communication system while texting, authors of paper [10] proposed a model that will predict and suggest grammatically more appropriate next word reducing the keystrokes required by the users. They have used a probabilistic language model based on N-gram for prediction.

In the paper [11], the author represents a new approach based on context features and machine learning to predict words. This method considered the problem as a learning-classification task. By using various feature selection techniques of machine learning, SVM (Support Vector Machin) with MI, X2 and more, author trained the word predictor. The experiment was done using several datasets, and compared to other similar works, the accuracy of predicting words using this method gives 91 % correct prediction.

People with disabilities may need a tool for communicating with each other. In this case, text generation is an essential activity to make the communication system easier for them. The objective of this research [12] is to upgrade the quality of a word prediction system for Brazilian Portuguese to reduce the time to write texts. In this research, authors have used the POS (Parts-of-Speech) from the previous words to predict the possible POS of the next words. The predicted words were generated from those anticipated POS and the information restrained in the lexicons. They have used the artificial neural networks, SVM, and regularized logistic models to predict word POS in Brazilian Portuguese, depending on the POS of the 1, 2, 3, or 4 previous words. Again, they have also discussed a meta-learning strategy for algorithm selection and a fusion algorithm to combine those. After the experiments, it was concluded that this approach could correctly predict 79.95% of the words with a maximum of 28.5% hit rate.

Again in paper [5], researchers have developed a model for predicting the next word of Bangla sentence using stochastic, i.e., N-gram language model such as unigram, bigram, trigram, and deleted interpolation and back-off models for auto-completing a sentence by predicting next suitable word. They have mentioned that the results of this research were promising, and it was also helpful for reducing misspelling.

Another paper [13] says that the use of artificial intelligence in predicting the next word can provide a restriction-free communication and understanding. In most word prediction algorithms, it is observed that authors have used a collection of words that shows likeness and directed to a fixed linear path. But analyzing the frequency of the words with pattern recognition of machine learning and adding new words into a local dictionary can improve the overall process. Over time, it upgrades and provides accurate results and can be refined with comparison using a cloud-based dictionary.

III. Methodology

A. Dataset Overview

We have assembled a large amount of data in Bangla language to assessment the new approach that we have proposed, and the total amount of gathered data is 170 thousand, which were collected from different sources. Table 1 shows the statistics of the data gathered from their origins.

Table 1. List of Collected Dataset

Data Collection Source	Total Words	Unique words
The Daily Prothom Alo Newspaper [7]	75,000	9,686
Bangla Academic Books [8]	45,000	5,075
BBC News Bangla [9]	50,000	8,239

After collecting the dataset from different sources, we have cleaned the dataset using the cleaning function, which was implemented to remove the unwanted objects ("", (), /,!),

English, and other language words from the Bangla text dataset. This cleaning function also helped to turn the initial dataset into a standard one so that it can be used later for other purposes. Afterward, 5 different datasets were created using the idea of n-gram which are uni-gram, bi-gram, tri-gram, 4-gram and 5-gram from the cleaned dataset. Figure 1 shows the structure of dataset cleaning and created the dataset. N-gram is a language model that assigns probabilities in a sequence of words, and as the n-gram language model is a clinging series of n items of a simple text, we have used it in this study [14]. Generally, the number of input words can be different each time when we want to predict the next word or words. Whenever the input word is a single one, output will always be a single word, which is called 1-gram or uni-gram [15]; when input will be two words and output will be a single word, it is called 2-gram or bi-gram [15]. Again, when input is three words and output will be a single word, it is called 3-gram or tri-gram [15] and uniformly for 4-gram and 5-gram. From our knowledge, n-gram language model has introduced up-to 4-gram, but we have initiated till 5-gram in this work. Because when the input length is more than 5 words, we only take the last 5 words as input and sent them to the trained 5-gram model. Usually the last 4 or 5 words are sufficient for understanding the dependency of the sequence. Now to understand better about this language model, suppose we have a Bangla sentence for instance,

বাংলা ভাষা আমাদের মায়ের ভাষা।

Uni-gram model presentation of this sentence is-

Input, X	Output, Y
বাংলা	ভাষা
ভাষা	আমাদের
আমাদের	মায়ের
মায়ের	ভাষা।

Bi-gram model presentation for this sentence is -

Input, X		Output, Y
X1	X2	
বাংলা	ভাষা	আমাদের
ভাষা	আমাদের	মায়ের
আমাদের	মায়ের	ভাষা।

Likewise, for 3-gram or tri-gram, it takes three input values and gives one corresponding output value like uni-gram and bi-gram and similarly for 4-gram, 5-gram.

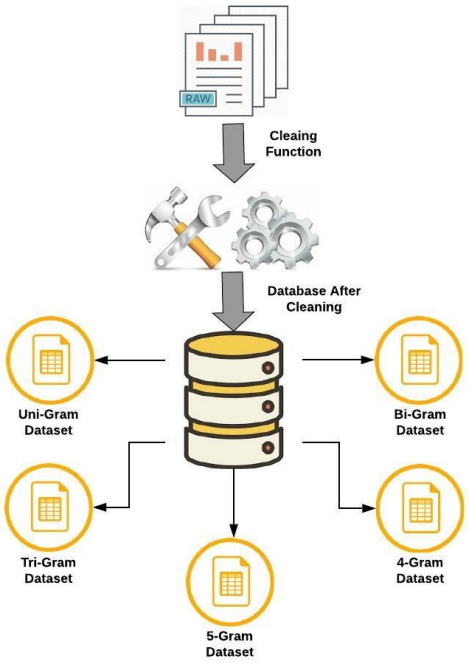

Figure 1: Structure of Dataset Processing

B. Implementation

In the basic n-gram language model, it measures the probability of possible next word, and whenever it gets all the probabilities for next possible word, it chooses the highest probability word and sets it as the next word (one or more). The next most likely value is calculated from a probabilistic view, and as we wanted to predict the next value from the input value বাংলা ভাষা, we have calculated the highest probability of the next word using equation (1).

Next word = Max (P (আমাদের | বাংলা ভাষা, P (মায়ের | বাংলা ভাষা), P (ভাষা | বাংলা ভাষা)) --------- (1)

Along with the satisfactory contribution of the n-gram in word prediction, there is a hindrance to the n-gram language model, and that is, n-gram language model cannot deal with issues arrived for zero probability caused by the absence of the expected next word in the dataset. Hence, most often model shows zero probability and cannot suggest the next word with highest probability, and the whole method get failed to predict the most appropriate value. N-gram also works inefficiently when the dataset is vast with very long sequence of data input or a considerable amount of N. Several methods, Back-off and Katz Back-off was applied to smooth the probability distribution by tuning n-gram with small count [17]. However, we have upgraded n-gram with Neural Network to solve the problem of n-gram because Neural Network can detect the pattern of input and suggest corresponding output [18] and in this work, we are dealing with Bangla corpus data to predict the next suitable words, and also it suggests a complete sentence. Hence, considering the dataset that we have used and our aim of this research, Neural Network, as well as RNN (Recurrent Neural Network), was also used because it works better with sequential data with dependencies among themselves to upskill that 5 different datasets.

RNN perpetuated state by using its' output as input from one to the next iteration, and this task of RNN is expressed by equation (2) where u is the weight, multiplied with the

current input x_t and w is the weight, multiplied with the previous output Y_{t-1} [19]. Figure 2 shows the structure of training the models with GRU based RNN.

$$Y_t = tanh\,(wY_{t-1} + ux_t) \text{--------- (2)}$$

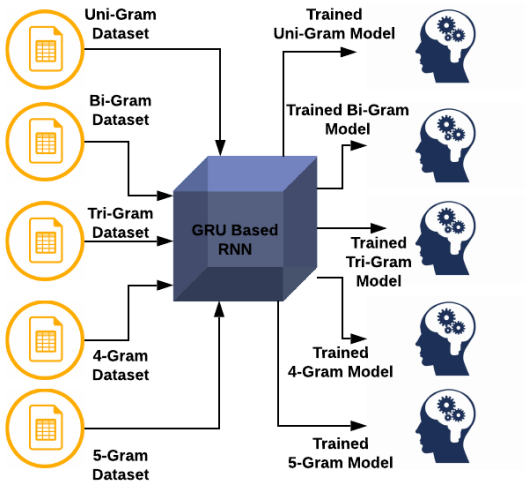

Figure 2: Architecture of training the models with GRU based RNN

But RNN also has a problem remembering the effect of the earlier layers in a long sequence, which means that even if the values of the parameters of the early layers change dramatically, the effect on the output is shallow, and this problem of RNN is studied as vanishing gradient problem.

Generally, two gating mechanism is used to solve the vanishing gradient problem; LSTM (Long Short Term Memory) and GRU (Gated Recurrent Unit) which allows us to solve the vanishing gradient problem. But we have used GRU in this study because it uses two gates such as update gate and reset gate where LSTM uses three gates: input, forget, and output [16]. Again, LSTM maintains an internal memory to remember the effect of the earlier layers where GRU doesn't need any extra memory, which makes it easier to implement and also requires less time to train the dataset. Hence, it makes GRU more efficient for a medium length sequence data than LSTM.

In GRU, update gate and reset gate are vectors that help to decide which information will pass through, and also they are trained to remember data from long ago without removing data that are irrelevant for prediction [17]. There are two mathematical notations, equation (3) and equation (4) of update gate and reset gate, which determines how much past data needed to be remembered and how much past data required to be forgotten [17]. In equation (3), z_t is the update gate which is calculated for time step t where $W^{(z)}$ is it's weight and h_{t-1} is the information of previous t-1 unit, which is multiplied by $U^{(z)}$. Again, r_t is the reset gate expressed in equation (4) which is calculated for time step t where $W^{(r)}$ is it's weight and h_{t-1} is the information of previous t-1 unit which is multiplied by $U^{(r)}$.

$$z_t = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1}) \text{--------- (3)}$$

$$r_t = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1}) \text{--------- (4)}$$

As shown in Figure 2:, we used GRU based RNN to train our previously created five datasets (Uni-gram to 5-gram) and built five corresponding models and Figure 3 shows the structure of the trained models which have five hidden layers named embedding_1 (Embedding), gru_1 (GRU), gru_2 (GRU), dense_1 (Dense) and dense_2 (Dense) with total parameters of 1,681,721 (params).

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 1, 50)             518550
_________________________________________________________________
gru_1 (GRU)                  (None, 1, 100)            45300
_________________________________________________________________
gru_2 (GRU)                  (None, 100)               60300
_________________________________________________________________
dense_1 (Dense)              (None, 100)               10100
_________________________________________________________________
dense_2 (Dense)              (None, 10371)             1047471
=================================================================
Total params: 1,681,721
Trainable params: 1,681,721
Non-trainable params: 0
```

Figure 3: Structure of layers in our training process

1) Word Prediction

After training all five datasets (Uni-gram to 5-gram), we now have 5 trained models for different length inputs. These models take different length word sequences as input and determine a single output, which is the next most likely word that should follow the input word sequence. Now, if the input word sequence length is one then it will be sent to train Uni-gram model as the model will take only one-word input and predict the most likely next word. Uniformly if the number of input words is two then the inputted words should be sent to trained bi-gram model as it takes two input words and predicts an output word. Likewise, for the rest of the trained models. Figure 4 represents the word prediction process for different length input using 5 trained models. There is an exception if the length of the inputted word sequence is higher than five. In such cases, only the last five words would be used in the trained 5-gram model to predict the next word. Because, in general, the last 4 or 5 words is enough to establish the dependency of the sequence.

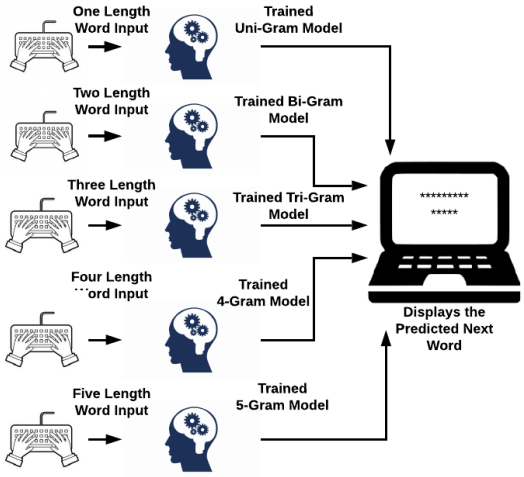

Figure 4: Word prediction from the trained models

2) Sentence Prediction

In our work, we not only predict the next most likely word but also suggest a full sentence from the given word sequence. To do that, we have used our previously

mentioned architecture of N-gram trained model trained by GRU based RNN. When we have the input values, we can predict the next value from the given sequence and then add the output (predicted word) with the input. So that we can predict furthermore words from the newly updated input, which makes it a complete sentence eventually. This process should be continued until the end of a sentence is determined. In Bangla language, the end of a sentence is determined by the use of punctuation, "।" for the normal statement, and "?" for a questioning statement. So the model will keep predicting the word sequence until the end punctuation of the sentence is found. Thus, the total output should be the suggested possible sentence.

IV. Result Analysis

To ratify the proposed approach, it's essential to run the experiments and analyzed the outcome earnestly. Hence, we have appraised our proposed approach on a corpus dataset training the five different models having identical structures until 1000 epochs (Figure 3).

Figure 5 and Figure 6 represents that, the trained Uni-gram model has an average accuracy of 32.17% and the average loss of 276.44% for our proposed approach, where the Bi-gram model has an average accuracy of 78.15% and the average loss of 53.36%. Again, Tri-gram has 95.84% accuracy on average and 8.52% loss on average for the same dataset used for Uni-gram and Bi-gram. Uniformly 4-gram and 5-gram show an average accuracy of 99.24% and 99.70% where they have an average loss of 2.04% and 1.11%, which indicates that the accuracy and loss level is improved according to the number of n is increased.

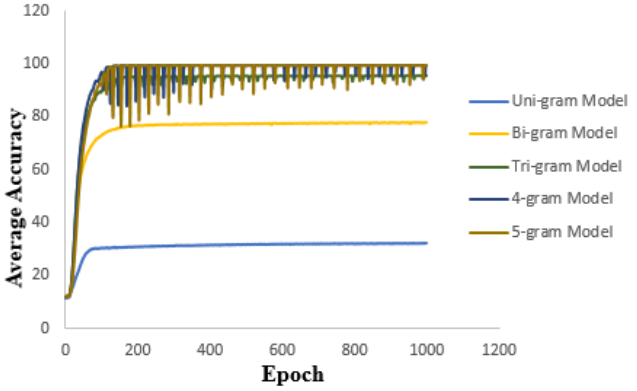

Figure 5: Graphical Representation of Average Accuracy of Trained Models in Percentage against 1000 Epochs

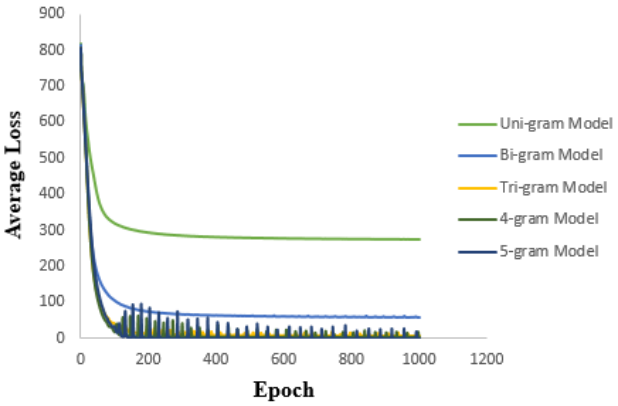

Figure 6: Graphical Representation of Average Loss of Trained Models in Percentage against 1000 Epochs

We have also compared the experimental result of our proposed approach with other approaches proposed by the researchers in paper [1] & [2] and found that, paper [1] has an accuracy of 88.20%, and paper [2] has an accuracy of 63.5% on average for their proposed method, where we have a maximum efficiency of 95.84%-99.70% on average for higher-order sequences. Figure 7 shows the comparison among the different approaches used in paper [1], paper [2] and this study.

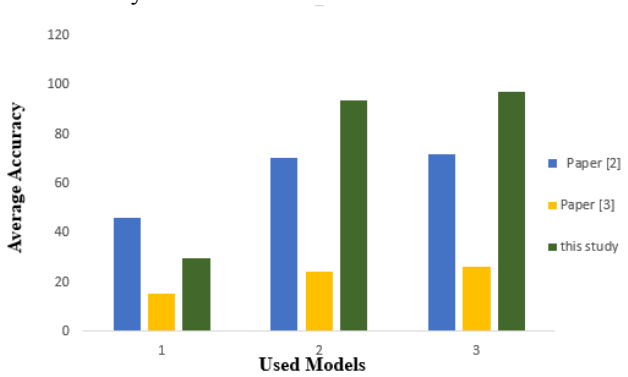

Figure 7: Comparison Chart of Average Accuracy

V. Conclusion

To predict the next most appropriate and suitable Bangla word (one or more) and sentence, GRU based RNN has shown a significant contribution to this research work. To justify the significance of using GRU based RNN, we have compared our proposed method with other methods that were used for Bangla and other languages by the researchers and got better accuracy among them (Figure 7). Although Uni-gram gives poor accuracy for our proposed work (32.17%), for higher-order sequences such as Tri-gram, 4-gram, and 5-gram, the accuracy rate is high (respectively 95.84%, 99.24%, and 99.70%). Again, the overall accuracy of this approach would be more impressive if we could use a larger dataset that we have already used in this work. Using Bangla corpus dataset was challenging as there is no readymade dataset for Bangla language, and we had to collect the dataset from different sources. In coming times, we will try to collect a large dataset to get better performance on GRU based RNN for Bangla next word and sentence prediction. Furthermore, this study will help as a tool for sustainable technologies in industry as its application is vast and it can be used in different sectors.

VI. References

[1] P. P. Barman and A. Boruah, "A RNN based Approach for next word prediction in Assamese Phonetic Transcription," *Procedia Comput. Sci.*, vol. 143, pp. 117–123, 2018.

[2] M. T. Habib, A. Al-Mamun, M. S. Rahman, S. M. T. Siddiquee, and F. Ahmed, "An Exploratory Approach to Find a Novel Metric Based Optimum Language Model for Automatic Bangla Word Prediction," *Int. J. Intell. Syst. Appl.*, vol. 10, no. 2, pp. 47–54, Feb. 2018.

[3] "What is Word Prediction?" [Online]. Available: http://www2.edc.org/ncip/library/wp/What_is.htm. [Accessed: 03-Aug-2019].

[4] S. Bickel, P. Haider, and T. Scheffer, "Predicting sentences using N-gram language models," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05*, 2005, pp. 193–200.R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[5] M. M. Haque, M. T. Habib, and M. M. Rahman, "AUTOMATED WORD PREDICTION IN BANGLA LANGUAGE USING

STOCHASTIC LANGUAGE MODELS," *Int. J. Found. Comput. Sci. Technol.*, vol. 5, no. 6, 2015.

[6] R. Makkar, M. Kaur, and D. V. Sharma, "Word Prediction Systems: A Survey," *Adv. Comput. Sci. Inf. Technol.*, vol. 2, no. 2, pp. 177–180.

[7] "Prothom Alo | Latest online Bangla world news bd | Sports photo video live." [Online]. Available: https://www.prothomalo.com/. [Accessed: 25-Aug-2019].

[8] "খবর, সর্বশেষ খবর, ব্রেকিং নিউজ, বিশ্লেষণ - BBC News বাংলা." [Online]. Available: https://www.bbc.com/bengali. [Accessed: 25-Aug-2019].

[9] "Bangla Academy Sangkhipto Bangla Avidhan (Bengali to Bengali Dictionary) ~ Free Download Bangla Books, Bangla Magazine, Bengali PDF Books, New Bangla Books." [Online]. Available: https://www.gobanglabooks.com/2017/08/bangla-academy-sangkhipo-bangla-avidhan.html. [Accessed: 25-Aug-2019].

[10] Dumbali, J & Nagaraja Rao, A. (2019). Real-time word prediction using N-grams model. International Journal of Innovative Technology and Exploring Engineering. 8. 870-873.

[11] Al-Mubaid, Hisham. (2007). A Learning-Classification Based Approach for Word Prediction. Int. Arab J. Inf. Technol. 4. 264-271.

[12] Palazuelos-Cagigas, Sira & Martín, José & Macias-Guarasa, Javier & García-García, J.C. & Cavalieri, Daniel & Bastos, Teodiano & Sarcinelli-Filho, Mário. (2011). Machine learning methods for word prediction in Brasilian Portuguese. Assistive Technology Research Series. 29. 424-431. 10.3233/978-1-60750-814-4-424.

[13] N. Ajithesh, "Artificial intelligence in word prediction systems," *Int. J. Adv. Res. Ideas Innov. Technol.*, vol. 4, no. 5, pp. 42–45, 2018.

[14] "n-gram", n.d. from https://en.wikipedia.org/wiki/N-gram. [Broder, Andrei Z.; Glassman, Steven C.; Manasse, Mark S.; Zweig, Geoffrey (1997). "Syntactic clustering of the web". Computer Networks and ISDN Systems. 29 (8): 1157–1166.

[15] "Introduction to Language Models: n-gram", n.d. from https://towardsdatascience.com/introduction-to-language-models-n-gram-e323081503d9. [CHAPTER DRAFT — | Stanford Lagunita. https://lagunita.stanford.edu/c4x/Engineering/CS-224N/asset/slp4.pdf]

[16] "neural network - When to use GRU over LSTM? - Data Science Stack Exchange." [Online]. Available: https://datascience.stackexchange.com/questions/14581/when-to-use-gru-over-lstm. [Accessed: 31-Jul-2019]

[17] Understanding GRU Networks - Towards Data Science. (n.d.). Retrieved July 31, 2019, from https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be

[18] "RPubs - Next Word Prediction using Katz Backoff Model - Part 2: N-gram model, Katz Backoff, and Good-Turing Discounting." [Online]. Available: https://rpubs.com/leomak/TextPrediction_KBO_Katz_Good-Turing?fbclid=IwAR0GoO0PGB66CD6gtupJhhQ7i1LJwZGDXutD9eAow3B5ry9YQBbu0dGk_xc. [Accessed: 02-Aug-2019].

[19] "How Predictive Analysis Neural Networks Work." [Online]. Available: https://www.dummies.com/programming/big-data/data-science/how-predictive-analysis-neural-networks-work/. [Accessed: 02-Aug-2019].

[20] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[21] E. A. Emon, S. Rahman, J. Banarjee, A. K. Das and T. Mittra, "A Deep Learning Approach to Detect Abusive Bengali Text," 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019.

[22] M. M. Hossain, M. F. Labib, A. S. Rifat, A. K. Das and M. Mukta, "Auto-correction of English to Bengali Transliteration System using Levenshtein Distance," 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019.

[23] M. D. Drovo, M. Chowdhury, S. I. Uday and A. K. Das, "Named Entity Recognition in Bengali Text Using Merged Hidden Markov Model and Rule Base Approach," 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019.

[24] E. Biswas and A. K. Das, "Symptom-Based Disease Detection System In Bengali Using Convolution Neural Network," 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, Malaysia, 2019.

[25] R. A. Tuhin, B. K. Paul, F. Nawrine, M. Akter and A. K. Das, "An Automated System of Sentiment Analysis from Bangla Text using Supervised Learning Techniques," 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 2019, pp. 360-364.

[26] J. Islam, M. Mubassira, M. R. Islam and A. K. Das, "A Speech Recognition System for Bengali Language using Recurrent Neural Network," 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 2019, pp. 73-76.