

NEXT WORD PREDICTION USING RECURRENT NEURAL NETWORK

Atharv Patil¹, Vaishnavi Dere², Pushkar Visave³, Sumit Patil⁴, Aniruddha S. Rumale⁵

^{1,2,3,4}Under Graduate Student Information Technology Sandip Institute of Technology and Research Centre, India.

⁵Professor Department of Information Technology Sandip Institute of Technology and Research Centre, India.

DOI: <https://www.doi.org/10.58257/IJPREMS32232>

ABSTRACT

Language modelling, a key task in Natural Language Processing (NLP), involves predicting the subsequent word and has a wide array of applications. We utilized Nietzsche's default text record to construct a model that predicts the next words after a user has input n letters. This model, developed with Recurrent Neural Network (RNN) and Tensor flow, comprehends n letters and forecasts the top words. Our objective was to predict 10 or more words in the shortest possible time. Thanks to the long short-term memory of the RNN, it can understand previous content and predict words, which aids users in forming sentences. This can improve users' lives and minimize risks. The model is also trained to comprehend and predict words in Hinglish. We introduce a Bi-Directional Long Short Term Memory (LSTM) network, a unique type of Neural Network. In this Recurrent Neural Network (RNN), our goal is to predict the next word for a given set of words in the model.

1. INTRODUCTION

When we contemplate the advancements brought about by Natural Language Processing (NLP), we often marvel at its extensive applications and its status as an undeveloped field of exploration. NLP, a pivotal component of artificial intelligence (AI), revolutionizes communication methods and amplifies the benefits derived from these interactions. One such benefit is evident in our daily texting habits, where the predictive text feature suggests the next word as we type, enhancing our texting experience. NLP's primary function is to predict subsequent words, a fundamental aspect of language modelling. As you delve into this paper, you comprehend each word based on the preceding one, indicating a sequence in thoughts. This sequence is crucial in understanding language and communication. However, Traditional Neural Networks often encounter limitations in storing large volumes of data. This led to the advent of Recurrent Neural Networks (RNN), networks equipped with loops that aid in information retention. These loops enable RNNs to remember previous inputs, making them ideal for tasks where context from earlier inputs is required. Consider a sentence like "The weather in Paris is quite unpredictable; hence, I always carry an umbrella." Here, to predict that the next word could be 'umbrella,' we need the context of 'Paris' and 'unpredictable weather,' which is stored and recalled by RNNs. However, for longer sentences or when the required context is far back, RNNs face challenges. To address this long-term dependency issue, Long Short Term Memory (LSTM) was introduced. LSTMs are designed to remember long-term dependencies by default and are therefore better suited for such tasks. Upon further research, we discovered that Bi-directional LSTM outperforms LSTM. Hence, we employed Bi-directional LSTM in this paper. Bi-directional LSTMs train on both sides of the input sequence - one in reverse order and the other from left to right. This dual training process adds another layer of context to the word, allowing it to fit into the correct context with words that come after and before it. This results in faster and more comprehensive learning and problem-solving.

2. LITERATURE REVIEW

A Next word prediction using the N-gram model has made the model more niche by only focusing on the Kurdish language. They have trained the model on the Kurdish text corpus. They had to face more difficulties because the Kurdish text corpus is very limited. To save time while typing the Kurdish language, the N-Gram model is utilized to predict the following word. When a user inputs a word, the system prompts them to type the next five words. That is based on the preceding written word or words, the suggested system will recommend the next five words. This model has an accuracy of 96.3% [1] A Vietnamese Language model used a recurrent neural network. Traditional Neural Networks can only understand words that they have seen before. The N-gram model is not suited for long-term dependencies. The model was trained on 24M syllables constructed from 1500 movie subtitles. In this paper, RNNs are explored for a Vietnamese language model. The following is a summary of the contributions: Building a Vietnamese syllable-level language model based on RNNs. Building a Vietnamese character-level language model based on RNNs. Extensive testing on a 24 million syllable dataset derived from 1,500 movie subtitles. Also, this model concludes that RNN based language model yields better results. The perplexity of 83.8% is thought to be

reasonable as this model Outstands the N-gram model in terms of results [2] A paper based on the Ukrainian Language analysed the next word Prediction model but it concentrates more on the Ukrainian Language. One main reason for working with a specific Ukrainian language is because of limited support for Ukrainian language tools. Their sequential character aids in completing the next-word guessing test successfully. The Markov chains produced the most accurate and timely results. The hybrid model produces adequate outcomes, but it is slow to implement the goal of this paper is to examine existing next- word prediction methods based on entered text and put them into the test in Ukrainian language material [3] In this research for Assamese Phonetic Transcription described a LSTM model for instant messaging, which is a type of RNN with the purpose of predicting the user's future words given a set of current. With an accuracy of 88.20 percent for Assamese text and 72.10 percent for phonetically Tran scripted the Assamese language, this model employs LSTM to predict the next word from a data set of Transcribed Assamese words. [4] Next word Prediction using RNN tried to create a model using the Nietzsche default text record that will predict the client's sentence after they have written 40 letters, the model will comprehend 40 letters and predict the top10 words using RNN neural organization and TensorFlow. Our goal in developing this model was to predict 10 or more words in the shortest amount of time possible. Because RNN has a long short-term memory, it can understand previous material and anticipate words, which can help users structure phrases. Letter-to-letter prediction is used in this technique, which means it predicts a letter after another to build a word. [5]

3. PROPOSED WORK

The proposed work aims to address the challenge of efficiently predicting the next word in a given sentence, with a focus on improving the user experience by providing real-time predictions. To achieve this, the authors have considered the needs of users while typing and have designed a model that can accurately forecast the next word based on the context of the input sequence.

- A. Dataset- The dataset used for this study consists of comments from two YouTube channels, which have been combined to form a single dataset. This dataset serves as the foundation for training and evaluating the model, and it provides a diverse range of texts that cover various topics and linguistic patterns. The proposed model uses Bi-directional Long Short Term Memory (LSTM) networks, which process input sequences in both forward and backward directions, allowing them to capture both past and future contexts. This enables the model to understand the relationships between words and their positions in a sentence, improving its ability to predict the next word.
- B. Pre-processing- Pre-processing is a vital step that removes unnecessary elements that can lower the model's performance and are not helpful for predicting the next word. This step prepares the data and filters out all irrelevant terms that might confuse the model or reduce its accuracy. For example, punctuation marks, numbers, stop words, and other noise can be removed from the data. We have 10 fields and 6508 records in our data, but we only use the title field for the next word prediction. The title field contains the headlines of news articles, which are short and informative. We need to eliminate some unwanted characters and words from it, as they can affect the model's accuracy negatively. For instance, we can remove words like "the", "a", "an", etc., as they do not contribute much to the meaning of the sentence. After that, we apply the Tokenization process, which assigns a unique id to each word and creates a word index.



Fig. 1. ARCHITECTURE DIAGRAM

This process converts the text into numerical values that can be fed into the model. The word index is a dictionary that maps each word to its corresponding id.

Many papers try to build a model to predict the next text, but only a few of them are effective, such as a work that uses SVM N-gram and RNN to predict the next code. This approach is useful, but it has some limitations, such as requiring a large amount of data and being prone to overfitting. A new algorithm like LSTM or Bi-directional LSTM might be able to produce better results for this problem statement, as they can capture long-term dependencies and learn from both past and future contexts. The current system has some limitations, which are as follows:

- Since we have to predict words that users think, the accuracy is quite low compared to other ML techniques. This is because human language is complex and ambiguous, and users might have different preferences and styles of writing. Therefore, it is hard to predict what word will come next in a given context.
 - Since the problem is complex, algorithms like SVM, Decision Tree, and others do not give good results and take more time to predict the outcomes. These algorithms are not suitable for sequential data, as they do not consider the order and relationship of words in a sentence. They also have high computational costs and memory requirements, which make them inefficient for large-scale data.
 - A text predictor needs as much linguistic information as possible to make strong predictions, so we have to train the neural network on new languages and data regularly. This is because language is dynamic and evolving, and new words and expressions are constantly being created and used by people. Therefore, we have to update our model frequently with new data and vocabulary to keep up with the changes in language use.
- C. LSTM VS Bi-directional LSTM- For a long time, there have been issues with sequence prediction. They are considered one of the most challenging challenges to solve in the data science industry. Long Short Term Memory networks, often known as LSTMs, have been discovered to be the most effective solution for practically all of these sequence prediction challenges thanks to recent developments in data science. We prioritize our appointments when we plan our day's schedule, right? We know which meeting could be cancelled to accommodate a possible meeting if we need to make some space for anything vital. It turns out that an RNN is incapable of doing so. LSTMs, on the other hand, perform tiny modifications to the data using multiplications and additions. Cell states are a system that transports information.

LSTMs can selectively recall or forget things in this way.

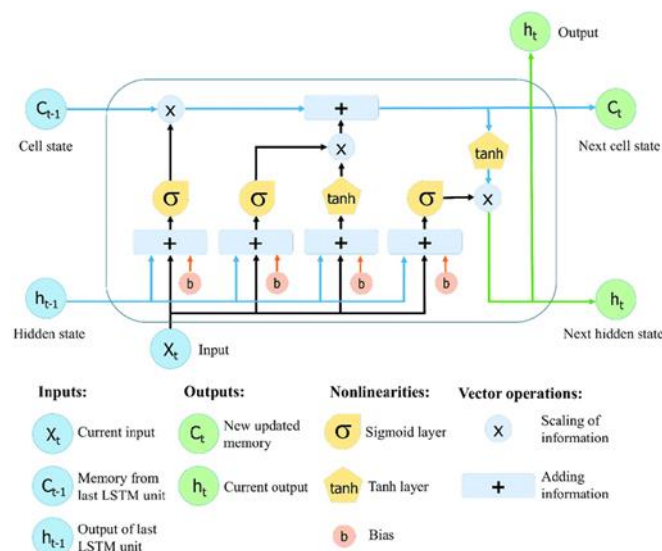


Fig.2.LSTMArchitecture diagram

Fig.2. shows it has four interacting layers with a unique method of communication. LSTM networks are a type of RNN architecture that "recalls" recently read values for a random time frame. There are specifically three gates in LSTMs control that gives how the information flow to and from their memories. The new data is fed to the memory using "input gate". The "forget gate" has control over how long particular values are held in memory. Activation of the block is affected by the "output gate" that manages the amount of the value contained in memory. These functionalities are shown in figure 2. The method of making any neural network have sequence information in both ways backward (future to past) or forward (ahead to future) is known as bi-directional long-short term memory (bi-lstm). The blank area in the line "boys go to..." cannot be filled. Still, when we have a future sentence like "boys come

out of school,” we can easily anticipate the previously blank space and have our model do the same thing, and bidirectional LSTM allows the neural network to do so. The hidden state is used by LSTM to preserve information from previously processed inputs. When you use Bi- directional LSTM, your inputs will be processed in two directions: one from the past to the future, and the other from the future to the past. The difference between this strategy and the LSTM that goes backward is that the LSTM that runs backward preserves information from the future, whereas the two hidden states combined maintain information from the past and future at any point in time. As a result, Bi-LSTM gave more precise results.

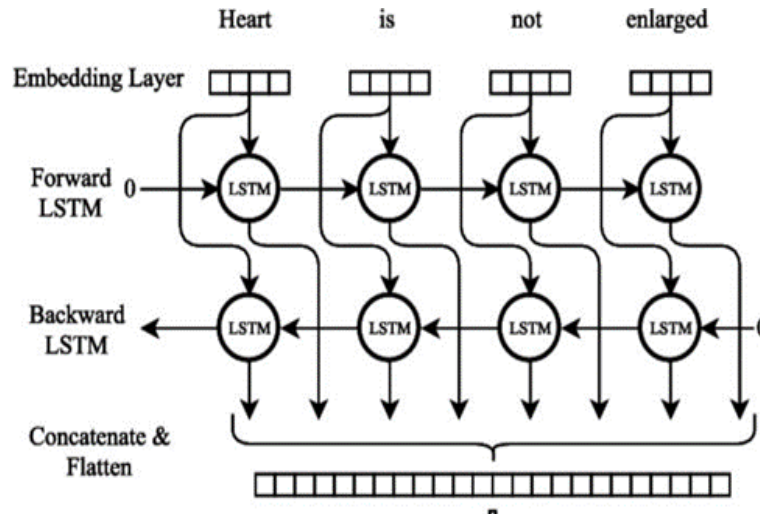


Fig.3.Bi-Directional LSTM Architecture Diagram

Fig.3. describes bidirectional model that consists of two input that is forward and backward. They effectively increase the amount of information available to the network that will help to improve the content available to the algorithm.

4. RESULTS

The table given below compares the accuracy between the existing models and the current model.

Performance of proposed model:

Table 1. Comparison Table between different models

Performance Analysis of Different Models			
Sr. No	Model Name	No Word Prediction	Accuracy
1	N-Gram Model[1]	The highest frequency for the top five words is projected based on the n-gram frequency.	92%
2	Long Short Term Memory (LSTM) [5]	It will count 10 words and give the user a list of them.	58.6%
3	Bi- Directional LSTM (Bi-LSTM) [Proposed Model]	Predicts N number of words as per the need	93%

Consider the N-Gram models mentioned in table 1, All 5 types of n-gram models are used whereas the model only works with a specific type of text corpus which is not suitable for all languages. When the system is unable to identify sufficient evidence to anticipate the following word, the N-gram is reduced. Our model works well and does not decrease the accuracy in any instance. The next model is a Long short-term memory (LSTM) in which the accuracy of the model itself is low and also because the only inputs it has seen are from the past, LSTM only saves information from the past. Our model outruns Long Short Term Memory in terms of accuracy and storing more information.

The BI-LSTM model shows good accuracy of 93%. A bidirectional LSTM differs from a standard LSTM in that the input flows in both directions. With a conventional LSTM, we may make input flow in one direction, either backwards or forwards.

We can have information flow in both directions with bi- directional input, maintaining both the future and the past. Due to this BI-LSTM proves to be the best model for next word prediction. This architecture offers numerous benefits in real-world issues, particularly in NLP. The major reason for this is that every component of an input sequence contains data from the past as well as the present. As a result, by merging LSTM layers from both directions, Bi-LSTM can create a more relevant output.

5. CONCLUSION AND FUTURE SCOPE

In the next word, Prediction has a veritably critical need at the moment and in the future itself. Transitional companies are trying this method because it makes them more user-friendly. Although there is still a lot of further exploration to be done in this particular field. Then, because it has memory cells to recall the one set, the bi-directional LSTM is employed to tackle the drawn-out dependency issue. In this model, our goal is to train and test an algorithm that is appropriate for this task and achieves high accuracy. This paper demonstrates how the system uses some mechanisms to predict and correct the next/target words, how the scalability of a trained system can be increased using the Tensor Flow closed-loop system, and how the system will decide that the sentence has more miss pelt words and how the system's performance can be improved using the perplexity concept.

- Something is rephrased when the same thing is written or stated in a new way, usually in a simpler and shorter form that clarifies the original meaning. Here our algorithm will predict more relatable words making it easier to form n number of sentences with the same meaning.
- The model will be trained on a music lyrics data set, this approach can help end-users predict then ext phrase in songs by developing lyrics and tunes, which is a major field in which this approach can help.
- Smart Compose expands on Smart Reply by predicting what you write next as you enter in the email body. The subject and prior email are encoded in this hybrid approach by averaging the word embedding in each field. Then, at each decoding step, combine those averaged embeddings and send them to the target sequence

6. REFERENCES

- [1] Hochreiter S., SchmidhuberJ.(1997).Lstm can solve hard long-time lag problems. Advances in neural information processing systems (1997), pp. 473-479
- [2] Bengio Y., SimardP., FrasconiP. (1994). Learning long-term dependencies with gradient descent is difficult IEEE transactions on neural networks, 5 (1994), pp.157-166.
- [3] Asma Rashid, H.K., Saeed, S. A. Rashid, T. A.(2021)Next word prediction based on the N-gram model for Kurdish Sorani and Kurmanji. Neural ComputApplic 33, 4547– 4566(2021).
- [4] A. F. Ganai and F. Khursheed,(2019) "Predicting next Word using RNN and LSTM cells: Statistical Language Modeling,"2019FifthInternationalConferenceonImage Information Processing (ICIIP), 2019, pp.469-474.
- [5] S. Siami-Namini, N. TavakoliandA. S. Namin, (2019) "The Performance of LSTM and BiLSTM in Forecasting TimeSeries,"2019IEEEInternationalConferenceonBig Data(BigData),2019,pp.3285-3292
- [6] R. Sharma, N. Goel, N. Aggarwal, P. Kaur, and C. Prakash,(2019)"Next Word Prediction in Hindi Using Deep Learning Techniques," 2019 International Conference on Data Science and Engineering (ICDSE), 2019, pp.55-60.
- [7] Kyuhyun Yeon, Kyunghan Min, Jaewook Shin, MyoungHo Sunwoo, Manabe Han, "Ego-Vehicle Speed Prediction Using a Long Short-Term Memory Based Recurrent Neural Network", International Journal of Automotive Technology, vol. 20, 2019, pp.713.
- [8] MinghuiWang,WenquanLiu,andYixionZhong,"Simple recurrent network for Chinese word prediction," Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan), Nagoya, Japan, 1993, pp.263-266.
- [9] Vikas Raunak,Vivek Gupta, and Florian Metze.2019. Effective dimensionality reduction for word embeddings. In Proceedings of the 4th Workshop on Representation Learning for NLP (Repl4NLP-2019), pages 235–243, Florence, Italy. Association for Computational Linguistics.
- [10] Bengio Y., Simard P., Frasconi P. (1994). Learning long-term dependencies with gradient descent is difficult IEEE transactions on neural networks, 5 (1994), pp. 157-166.
- [11] Asma Rashid, H.K., Saeed, S.A. Rashid, T.A. (2021) Next word prediction based on the N-gram model for Kurdish Sorani and Kur Manji. Neural Compute Applic 33, 4547–4566 (2021).
- [12] F. Ganai and F. Khursheed, (2019)" Predicting next Word using RNN and LSTM cells: Statistical Language Modeling," 2019 Fifth International Conference on Image Information Processing (ICIIP), 2019, pp. 469-474
- [13] S. Siami-Namini, N. Tavakoli and A. S. Nami, (2019)" The Performance of LSTM and BiLSTM in Forecasting Time Series," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 3285-3292
- [14] R. Sharma, N. Goel, N. Aggarwal, P. Kaur, and C. Prakash, (2019)" Next Word Prediction in Hindi Using Deep Learning Techniques," 2019 International Conference on Data Science and Engineering (ICDSE), 2019, pp. 55-60.

- [15] Kyuhyun Yeon, Kyunghyang Min, Jaewook Shin, Myoungho Sun woo, Manabe Han," Ego-Vehicle Speed Prediction Using a Long Short-Term Memory Based Recurrent Neural Network", International Journal of Automotive Technology, vol. 20, 2019, pp. 713.
- [16] Minghui Wang, Wenquan Liu, and Yixion Zhong," Simple recurrent network for Chinese word prediction," Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan), Nagoya, Japan, 1993, pp. 263-266
- [17] M. K. Sharma, S. Sarcar, P. K. Saha and D. Samanta, "Visual clue: An approach to predict and highlight next character," 2012 4th International Conference on Intel- ligitent Human Computer Interaction (IHCI), Kharagpur, 2012, pp. 1-7, doi: 10.1109/IHCI.2012.6481820.
- [18] Zhong, "Simple recurrent network for Chinese word pre- diction," Proceedings of International Confer- ence on Neural Networks (IJCNN-93-Nagoya, Japan), Nagoya, Japan, 1993, pp. 263-266 vol.1, doi: 10.1109/IJCNN.1993.713907.
- [19] Y. Ajioka and Y. Anzai, "Prediction of next alpha- bets and words of four sentences by adaptive junc- tion," IJCNN-91-Seattle International Joint Conference on Neural Networks, Seattle, WA, USA, 1991, pp. 897 vol.2-, doi: 10.1109/IJCNN.1991.155477.
- [20] Joel Stremmel, Arjun Singh. (2020). Pretraining Fed- erated Text Models for Next Word Prediction using GPT2
- [21] Sharma Y, Bindra JS, Aggarwal K, Dahiya N (2019) Word prediction and sentence completion. IJSRD—Int J Sci Res Dev 7(03)
- [22] Leshner GW, Moulton BJ, Higginbotham DJ (1999) Effects of ngram order and training text size on word prediction. In: Proceedings of the RESNA'99 annual conference, pp 52–54
- [23] Carlberger A, Carlberger J, Magnuson T, Sharon Hunnicutt M, Zuelos-Cagigas SP, Navarro A (1997) Profet, a new generation of word prediction: an evaluation study. In: ACL-97 workshop on natural language processing for communication aids
- [24] Bjorck J, Gomes CP, Selman B, Weinberger KQ (2018) Understanding batch normalization. In: Proceeding neural information processing system, Montreal, QC, Canada, pp 7694–7705
- [25] Bhuyan MP, Sarma SK, Automatic formation, termination and correction of assumes word using predictive and syntactic NLP. In: Proceedings of the international conference on communication and electronics systems
- [26] Palangi H et al (2016) Deep sentence embedding using long short-term memory net works: analysis and application to information retrieval. IEEE/ACM Trans Audio, Speech, Language Proc 24(4):694–707. <https://doi.org/10.1109/TASLP.2016.2520371>
- [27] Socher R, Bengio Y, Manning CD (2012) Deep learning for NLP (without magic). Tutorial Abstracts ACL 2012:5
- [28] Young T, Hazarika D, Poria S, Cambria E (2018) Recent trends in deep learning based natural language processing. IEEE Comput Intell Mag 13(3):55–75
- [29] Alshemali B, Kalita J (2019) Improving the reliability of deep neural networks in NLP: a review. Knowl-Based Syst. <https://doi.org/10.1016/j.knosys.2019.105210>.