Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, May : 2024

Next Word Prediction Using Recurrent Neural Network

Mr. Atharv Patil[1], Research Scholar, Dept. of Information Technology, Sandip Institute of Technology and Research Centre (SITRC), Nashik.

Miss. Vaishnavi Dere[2], Research Scholar, Dept. of Information Technology, Sandip Institute of Technology and Research Centre (SITRC), Nashik.

Mr. Pushkar Visave[3], Research Scholar, Dept. of Information Technology, Sandip Institute of Technology and Research Centre (SITRC), Nashik.

Mr. Sumit Patil[4], Research Scholar, Dept. of Information Technology, Sandip Institute of Technology and Research Centre (SITRC), Nashik.

Dr. Aniruddha S. Rumale[5], Professor, Dept. of Information Technology, Sandip Institute of Technology and Research Centre (SITRC), Nashik.

Abstract

Language modelling, a fundamental task in Natural Language Processing (NLP), involves predicting subsequent words and finds applications across various domains. In our research, we leveraged Nietzsche's default text corpus to construct a predictive model that anticipates the next words based on a user's input of n letters. Our model, developed using Recurrent Neural Networks (RNNs) and TensorFlow, effectively comprehends these initial n letters and generates top word predictions. Our primary objective was to predict at least 10 words within the shortest possible time frame. The RNN's long short-term memory (LSTM) architecture enables it to retain context from previous content, facilitating accurate word predictions. This capability assists users in constructing coherent sentences, thereby enhancing their overall experience and reducing potential risks. Notably, our model is also trained to comprehend and predict words in Hinglish, a unique blend of Hindi and English. Furthermore, we introduce a novel type of neural network: the Bi-Directional Long Short-Term Memory (LSTM) network. Within this architecture, our goal remains consistent—to predict the next word given a specific set of preceding words within the model.

Our research contributes to advancing language modelling techniques and has practical implications for improving user interactions with NLP systems.

Keywords: Long Short-Term Memory (LSTM), Bi- Directional LSTM (BI- LSTM), Recurrent Neural Networks

I. INTRODUCTION

Natural Language Processing (NLP) has witnessed remarkable advancements, captivating researchers with its diverse applications and its status as an evolving field of exploration. As a pivotal component of artificial intelligence (AI), NLP transforms communication methods, enhancing the benefits derived from human-computer interactions. A tangible manifestation of this impact lies in our everyday texting routines, where predictive text features seamlessly suggest subsequent words as we type, elevating our overall texting experience.

At the core of NLP lies the fundamental task of predicting subsequent words—a critical aspect of language modelling. As you delve into this research paper, consider each word as a building block, intricately connected to its predecessor. This sequential arrangement plays a pivotal role in comprehending language and facilitating effective communication.

However, Traditional Neural Networks grapple with limitations when it comes to handling large volumes of data. To address this challenge, Recurrent Neural Networks (RNNs) emerged, equipped with loops that aid in retaining information from previous inputs. These loops empower RNNs to recall context, making them well-suited for tasks requiring historical context.

Imagine a sentence like: "The weather in Paris is quite unpredictable; hence, I always carry an umbrella." To predict that the next word could be 'umbrella,' we rely on the context of 'Paris' and the notion of 'unpredictable weather,' stored and retrieved by RNNs. Yet, for longer sentences or distant context dependencies, RNNs encounter limitations.

Enter Long Short-Term Memory (LSTM), designed explicitly to handle long-term dependencies. LSTMs excel in retaining context, making them ideal for such tasks. Our research further revealed that Bi-directional LSTMs outperform their unidirectional counterparts. Consequently, we adopt Bi-directional LSTMs in this paper. These networks train on both ends of the input sequence—simultaneously in reverse order and from left to right. This dual training process enriches the word's context, enabling seamless integration within the broader linguistic context. The result: accelerated and comprehensive learning, enhancing problem-solving capabilities.

II. 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 Transcripted 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 top 10 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]

III. 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, we 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

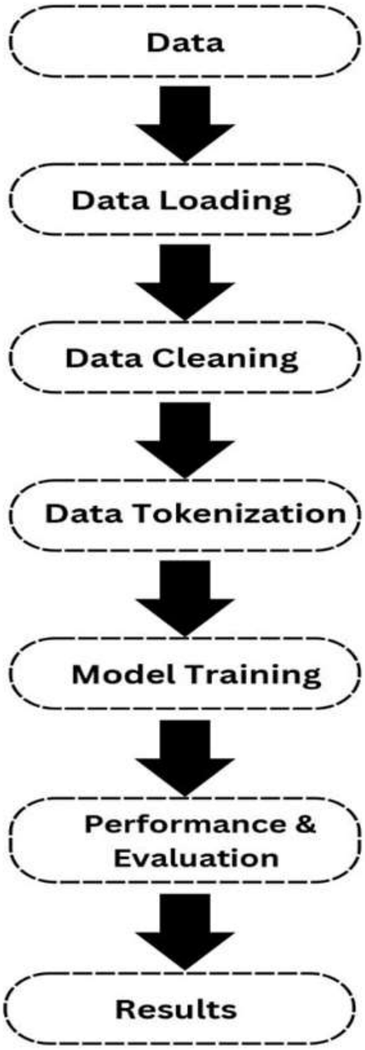
The dataset used for this study consists of comments from two YouTube channels, which have been combined to form a single dataset. This dataset serves as the foundation for training and evaluating the model, and it provides a diverse range of texts that cover various topics and linguistic patterns.

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. 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.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, May : 2024



Fig, 1. ARCHITECTURE DIAGRAM

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:

1. Challenges in Word Prediction Accuracy:

- Predicting user-intended words is inherently challenging due to the complexity and ambiguity of human language.

- User preferences and writing styles vary, making it difficult to accurately anticipate the next word in a given context.

2. Limitations of Traditional Algorithms:

- Algorithms like SVM and Decision Trees perform poorly in word prediction tasks.

- They lack consideration for word order and relationships within sentences.

- High computational costs and memory requirements hinder their scalability for large-scale data.

3. Linguistic Adaptability and Model Maintenance:

- To enhance predictions, neural networks require continuous training on diverse languages and evolving data.

- Language evolves, introducing new words and expressions.

- Regular model updates with fresh data and vocabulary are essential to stay current with language changes.

C. LSTM VS Bi-directional LSTM

Sequence prediction has long posed significant challenges in the data science domain. Among these challenges, Long Short-Term Memory networks (LSTMs) have emerged as a highly effective solution, fuelled by recent advancements in data science. LSTMs address a wide range of sequence prediction tasks.

Consider our daily schedules: we prioritize appointments, allocate time for meetings, and even anticipate potential adjustments. However, standard Recurrent Neural Networks (RNNs) fall short in this regard. LSTMs, in contrast, excel by subtly modifying data through multiplicative and additive operations. Their ability to retain context via cell states enables them to transport crucial information.

Furthermore, our research reveals that Bi-directional LSTMs surpass their unidirectional counterparts. By training on both ends of an input sequence—simultaneously in reverse order and from left to right—Bi-directional LSTMs enhance context comprehension and accelerate learning, making them a valuable tool in sequence prediction tasks. LSTMs can selectively recall or forget things in this way.

D. Selective Memory and Contextual Adaptability in LSTMs

While LSTMs excel at retaining context, their true power lies in their selective memory. These networks can selectively recall or forget information based on the task at hand. Imagine a conversation where context shifts rapidly—a user discussing travel plans, then transitioning to a recipe, and finally mentioning a favourite book. LSTMs adapt seamlessly, retaining relevant context while discarding irrelevant details.

Bi-directional LSTMs, with their dual training process, enhance this adaptability. By considering both past and future context, they create a holistic understanding of the sequence. This nuanced comprehension allows for faster learning and more accurate predictions.

In summary, LSTMs—both unidirectional and bi-directional—serve as indispensable tools in unravelling the intricacies of language, enabling us to predict and communicate effectively.

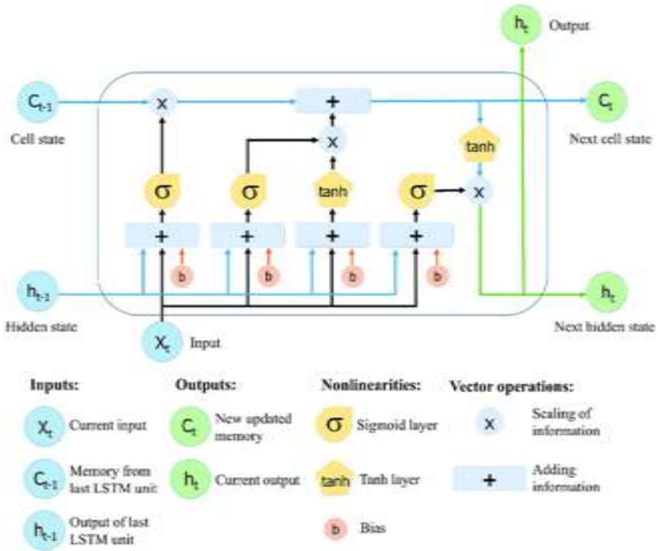
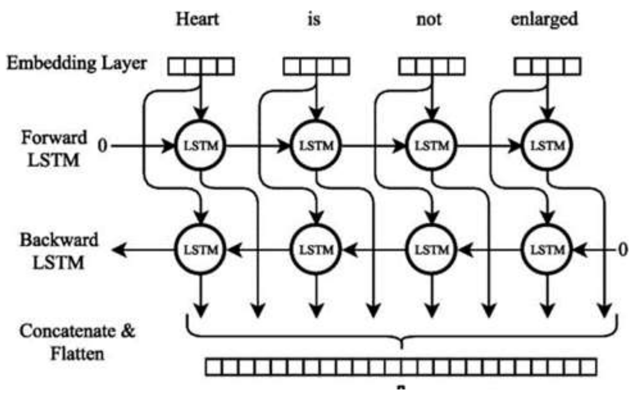


Fig.2.LSTMArchitecturediagram

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 "in- put 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 for- wards (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.



Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, May : 2024

Fig.3.Bi-DirectionalLSTMArchitectureDiagram

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 helps to improve the content available to the algorithm.

IV. RESULTS

A. Comparative Analysis and Model Performance

This section presents a comparative analysis of various predictive models, highlighting the superior performance of the proposed Bi-Directional LSTM (BI-LSTM) model. The comparison is based on word prediction accuracy, and the results are summarized in the following table:

Table 1: Comparative Accuracy of Predictive Models

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

Consider the N-Gram models mentioned in table1, all 5 types of n-gram models are used where as 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.

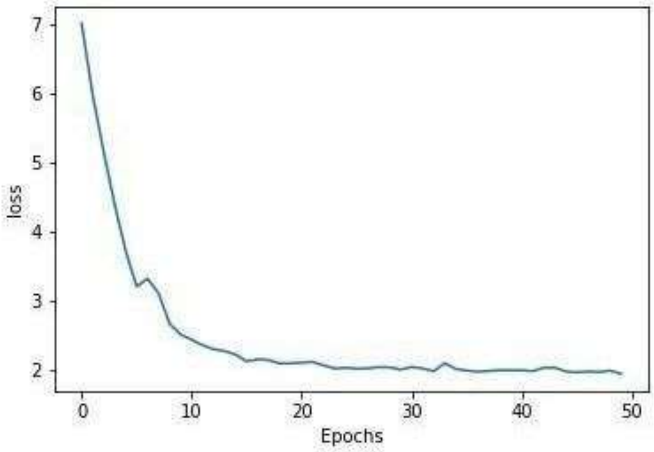
The next model is a Long short-term memory (LSTM) in which the accuracy of the model itself is low and also. Because the only inputs it has seen are from the past, LSTM only saves information from the past. Our model out runs Long Short-Term Memory in terms of accuracy and storing more information. The 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.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, May : 2024

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.

B. Loss and Accuracy Visualization

To further illustrate the effectiveness of the BI-LSTM model, we have included graphical representations of loss and accuracy over training epochs. These plots demonstrate the model's consistent performance and convergence over time.



Fig,4. Loss

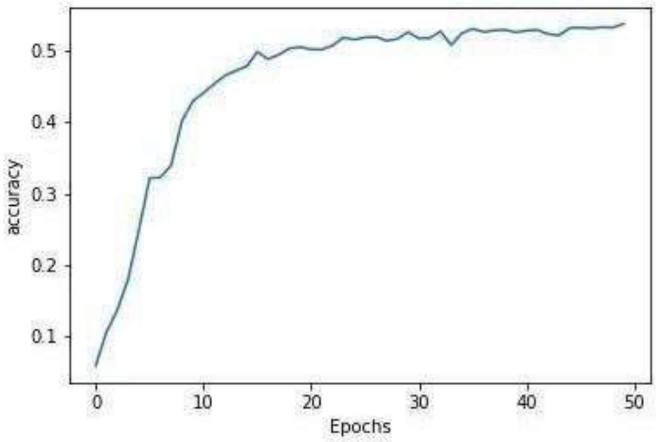


Fig.5. Accuracy

C. Output Demonstrations

Screenshots of the model's output showcase its predictive capabilities in practical scenarios. These visual aids provide tangible evidence of the model's functionality and its potential application in various NLP tasks.

Fig.6. Output

```
Enter your line: aap
Input Text: aap
1/1 [==============================] - 0s 17ms/step
Predicted Next Words:
1. ki
2. ka
3. bahut
Select a word (1, 2, or 3): 3
Selected Word: bahut
Completed Sentence: aap bahut
```

Fig.6.1. Output

```
Enter your line: hi
Input Text: hi
1/1 [==============================] - 1s 638ms/step
Predicted Next Words:
1. mam
2. apkaa
3. maam
Select a word (1, 2, or 3): 1
Selected Word: mam
Completed Sentence: hi mam
```

V. 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.

VI. 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] BengioY.,SimardP.,FrasconiP.(1994). Learning long-term dependencies with gradient descent is difficult IEEE transactions on neural networks, 5 (1994), pp.157-166.

[3].AsmaRashid,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,andC.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].KyuhyunYeon, 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,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.

[9].VikasRaunak,Vivek Gupta, and Florian Metze.2019. Effective dimensionality reduction for word em-beddings. 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 Applica 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- ligent Human Computer Interaction (IHCI), Kharagpur, 2012, pp. 1-7, doi: 10.1109/IHCI.2012.6481820.

[18] Yixion 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