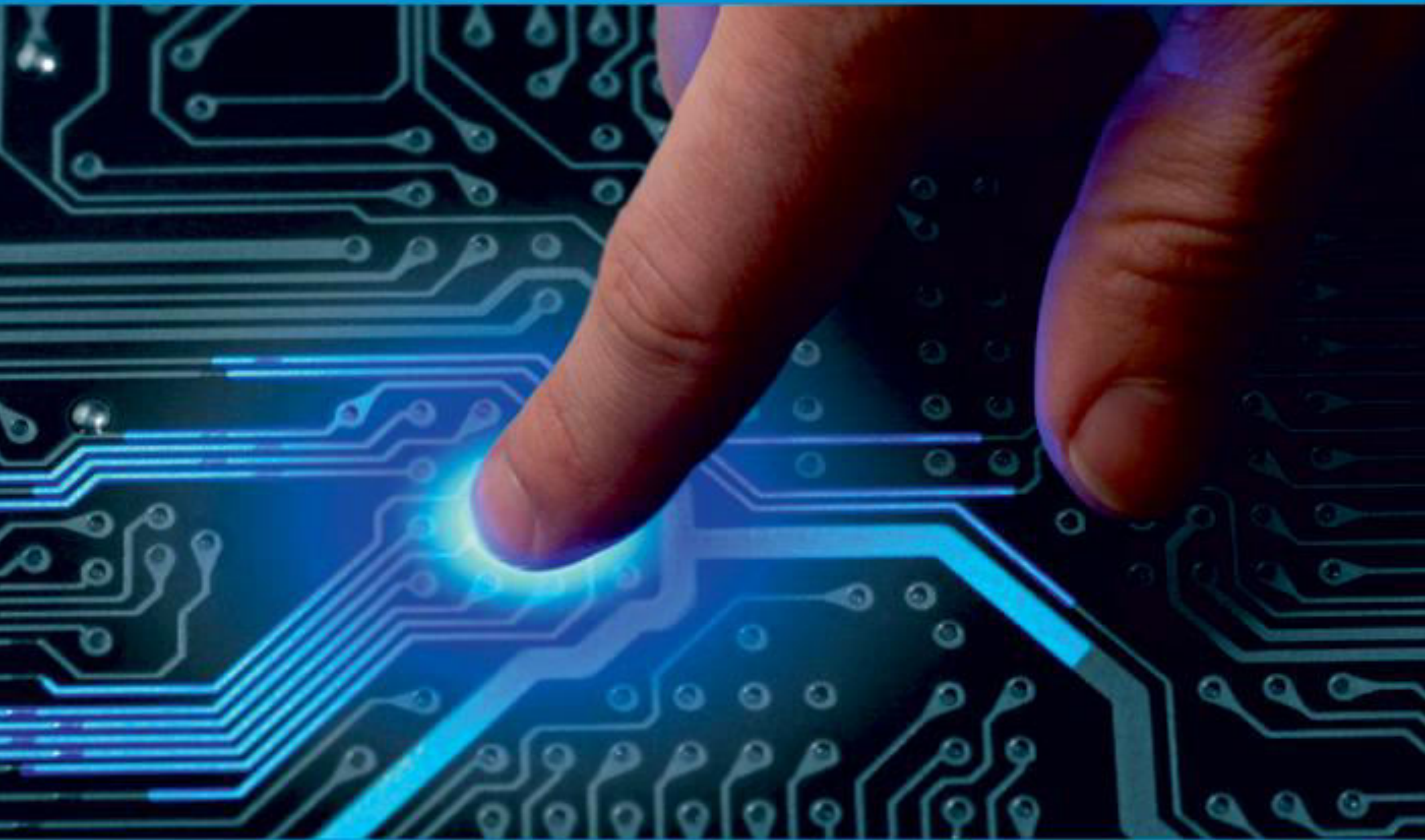
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 5, May 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

Next Word Prediction Using RNN & LSTM

Ganji Charan Tej[1], Eluri Sai Kavya[2], Bojjagani Rahul Rao [3], Battineni Sai Kumar [4], Maddala Jeevan Babu[5]

U.G. Student, Department of Computer Science and Engineering, VasireddyVenkatadri Institute of Technology, Nambur, Andhra Pradesh, India [1,2,3,4]

Assistant Professor, Department of Computer Science and Engineering, VasireddyVenkatadri Institute of Technology, Nambur, Andhra Pradesh, India [5]

ABSTRACT_ Almost all smartphone keyboards have next word prediction features, and Google employs this capability as well. As a result, the keyboard function of our smartphones is also filled with data to correctly guess the next word. Python is used to train a Recurrent Neural Network (RNN) for Next Word Prediction. In order to anticipate the output of a layer, RNN uses the notion of preserving the layer's output and feeding it back to the input. Using RNN's Long Short Time Memory (LSTM) and letter-to-letter prediction, it understands past text and predicts words that may be helpful to the user in constructing phrases. An RNN can handle sequential data, accepting both the current input data and the inputs that were previously received. RNNs have an internal memory that allows them to remember prior inputs. The goal of this model was to predict as many words as feasible in the shortest amount of time possible. Numpy and Keras libraries will be used in Python for our next word prediction model. Final output is the set of anticipated words for the supplied sentence.

1.INTRODUCTION

Text entering is made easier with the help of word prediction software. In mobile and tablet keyboards, it has become a popular choice. ' Based on the frequency and context of words in a sentence, word prediction software can help reduce the amount of keystrokes needed to type. Spell checking while you type, speech synthesis, and shortcuts for commonly used terms may also be included in these applications. A word prediction tool comes in handy if you type slowly or have trouble spelling or coming up with the right words to utilise in a sentence.

Speculates on the next word in a given sentence to use as a starting point. Every time we try to enter a message, a recommendation pops up in an attempt to forecast the next word we'll type. Predicting the next word is the primary goal in this method of prediction. Using Recurrent neural networks (RNNs), we have made some progress here. Predicting the next word accurately, efficiently, and quickly is a delicate balancing act.

An assistive technology tool for writing, called word prediction, recommends words to children as they enter them in their keyboards. There is a good chance that you've already used it when sending text messages from your smartphone. However, it's possible that you are ignorant of the process used to forecast the words.

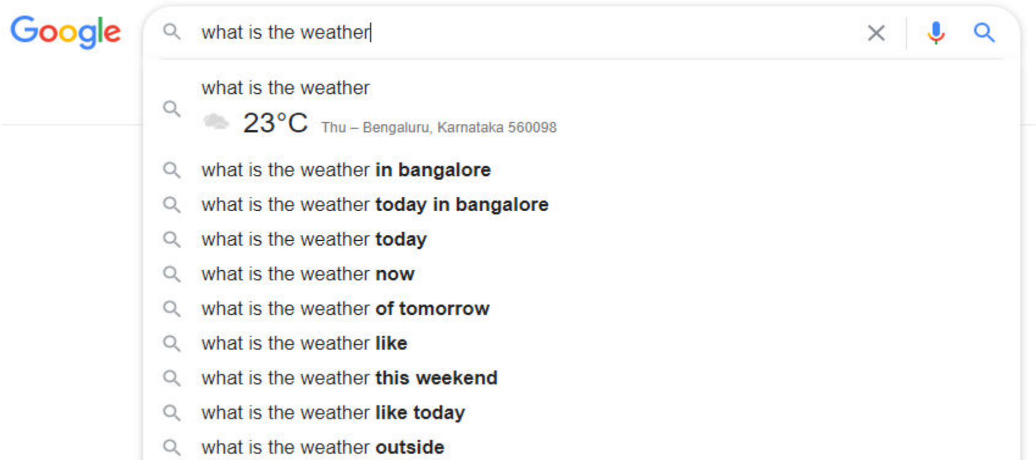


Figure 1: Word Prediction in Google search Engine

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

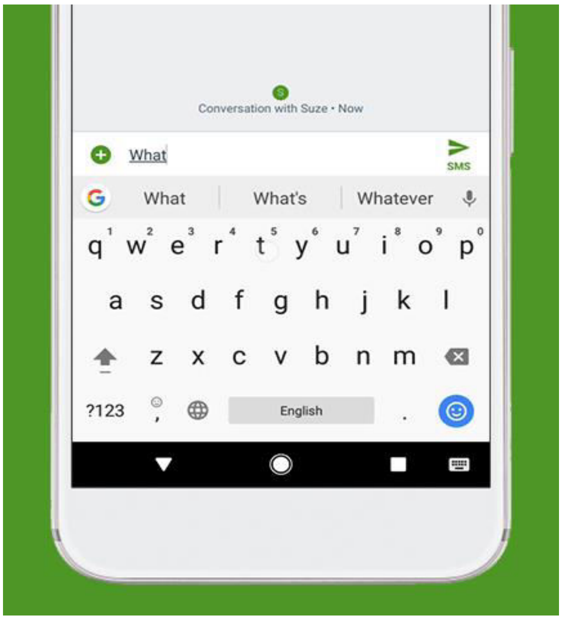


Figure 2: Word Prediction pop-up in mobile phone while typing/texting

Word Prediction or Text Generation is a traditional ideation problem in Machine Learning and Deep Learning, therefore it's effectively a language modelling problem. Recurrent Neural Network, or RNN, is being used to predict words in this case. Actually, Neural Networks are claimed to be based on the way our brains work. As a result, they can provide significant insights and useful applications on a variety of levels. Word or text prediction can be handled by using a Recurrent Neural Network (RNN).We'll talk about how RNNs lack the ability to retain information over the long term, which is referred to as Long-term Dependency. To solve this issue, LSTM (Long short-term memory) is the answer. To solve this issue, we've created a custom-built architecture.
RNN and LSTM (advancement model of RNN) are used in this project as an overview.

II.LITERATURE SURVEY

Numerous fields of study have seen significant progress thanks to machine learning methods based on artificial neural networks (ANNs). Accelerated by the use of Deep Learning (DL) approaches, which are ANNs with multiple layers, as well as GPU technology advancements. In numerous disciplines, including as object detection, image processing, computer vision, audio recognition, natural language processing (NLP), character recognition, and signature verification, deep learning algorithms have outperformed state-of-the-art methodologies. DL was built on McCulloch and Pitts' 1943 ANN proposal, but it only gained serious traction in 2012, when it was widely adopted. Due to the fact that they continuously obtain a local optimal solution, multilayer neural networks have proved ineffective. In addition, the processing power of big data has improved so much that interest in multi-layered neural networks hasn't been as high for a long time. Pre-training and fine-tuning, respectively, were introduced by Hinton et al. in 2006 to effectively train DL. As a result of this, interest in Digital Literacy began to rise. When Krizhevsky et al. improved the ImageNet competition's Top-5 error rate from 26.2% to 15.3% in 2012, it was a major breakthrough in object recognition. As a result of this achievement, interest in DL has skyrocketed among the academic community. Many technology businesses, in addition to academic institutions, sponsor the development of DL techniques. Researchers working in the field of deep learning (DL) are able to use frameworks established by businesses like Google, Facebook, Microsoft, and NVIDIA as open source software. Many layers, including Fully Connected (FC), Dropout and Pooling, are responsible for the success of Deep Learning. DL uses the backpropagation algorithm to show how a machine's internal parameters should be changed to compute the representation between layers in order to identify intricate structure in massive data sets. These diverse architectures include the Deep Neural Network (DNN), Convolutional Neural Network (CNN) and the Deep Belief Network (DBN), as well as the Auto-Encoding (SAE) and Recurrent Neural Network (RNN) models (RNN). Even while these frameworks have the same basic structure, there are some variances [11].

In 1943, Warren McCulloch and Walter Pitts[1] developed a mathematical model of neural networks based on threshold logic algorithms. This was the beginning of the development of artificial neural networks (ANNs). It was because of this concept that research began to be divided into two distinct streams. Biological processes and neural

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

networks were two different approaches to artificial intelligence. Finite automata and neural networks were the result of this research.

An important paper on how neurons might function was written by neurophysiologist Warren McCulloch and mathematician Walter Pitts in 1943. A rudimentary neural network was built using electrical circuits to depict how neurons in the brain might function.
An important idea in human learning is the fact that brain pathways are strengthened each time they are utilised, as Donald Hebb explained in The Organization of Behavior in 1949. A link between two nerves can be strengthened if they both fire at once.

Finally, in the 1950s, computers became advanced enough to simulate a hypothetical neural network. In the beginning, Nathaniel Rochester, a researcher at IBM's research labs, made a significant contribution. It didn't go well for him in his first effort, as he found out.
Stanford's Bernard Widrow and Marcian Hoff created the "ADALINE" and "MADALINE" models in 1959. Abbreviations like "Multiple ADAptiveLINear Elements" (MALE) are a staple at Stanford. Binary patterns were used to construct ADALINE in order to forecast what would come next while listening to phone lines that were streaming bits. MADALINE was the first neural network to use an adaptive filter to reduce phone line echoes to solve a real-world problem. Despite its age, the system is still being used in commercial settings, just like air traffic control systems.

Weight Change = (Pre-Weight line value) * (Error / (Number of Inputs)) was established by Widrow& Hoff in 1962, and it evaluates the value before the weight adjusts it. While one perceptron may have a large mistake, the weight values can be adjusted to transfer it to other perceptrons, or at the very least, to the adjacent perceptrons. Even if the line preceding the weight is 0, this rule will still produce an error. However, this will ultimately rectify itself. It is possible to erase the inaccuracy by distributing it evenly among all of the weights.
Despite the neural network's later success, von Neumann architecture dominated the computing scene and neural research was left behind. John von Neumann himself proposed utilising telegraph relays or vacuum tubes to mimic neurological processes.
The same period also saw the publication of a research suggesting that the single-layered neural network couldn't be extended to a multilayered neural network. A learning function that was essentially defective was used by many people in the field since it could not be differentiated throughout the full line. Because of this, research and funding have decreased significantly.

Because of this, there was an exaggeration in the potential of neural networks, especially in light of practical technology available at the time. In the end, promises were not kept, and philosophical questions sparked terror. The impact of "thinking machines" on humans has long been debated, and some of these theories are still relevant today.
Having a computer that can write its own software is a fascinating concept. A self-reprogramming version of Microsoft's Windows 2000 would have the ability to fix the hundreds of mistakes made by its programming workforce. These concepts sounded great, but putting them into action proved to be quite challenging. In addition, the architecture of von Neumann was becoming increasingly popular. For the most part, there were just a few breakthroughs in the discipline.

In 1972, Kohonen and Anderson independently built a similar network, which we will go into greater detail about later on. Their concepts were described using matrix mathematics, but they didn't understand they were developing an ADALINE circuit array. Instead of activating a single output, the neurons are designed to stimulate several.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

III.PROPOSED SYSTEM

Proposed work flow will be as following:

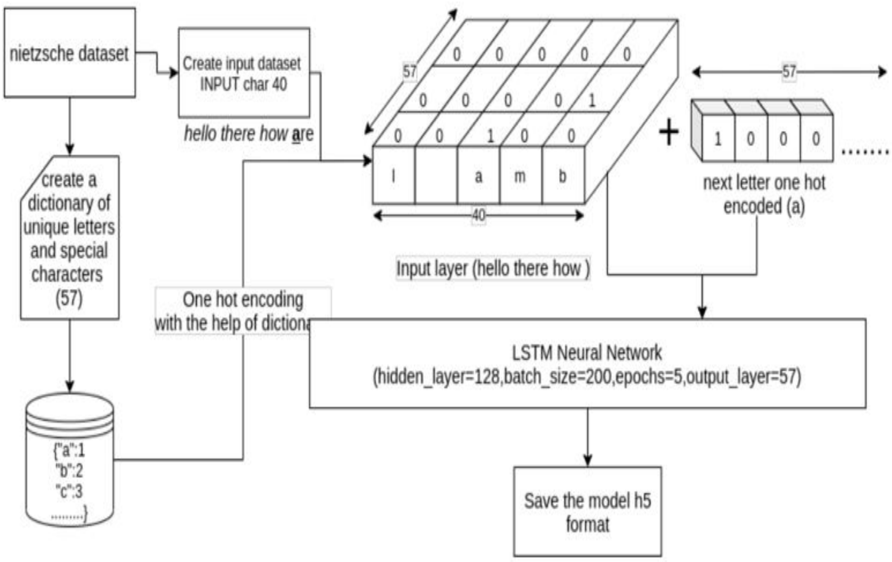


Figure 1: Proposed work flow

Fig. 1 shows an example of a flexible model that can assist users in detecting the next word while understanding their vocable in a quick and effective manner, such that the user must input letters before it sends this letter to LSTM NN and predicts N numbers of letters.

As depicted in the picture above, this statement will be processed by an LSTM Neural Network after receiving input data letters. In the Letter LSTM displayed above, each letter is learned and a score is created for the next letter. Finally, it will predict a word letter by letter using the same LSTM that was used to process the previous score.

The following is a breakdown of our neural network development process and how it was implemented using the Tensorflow library:

To bits and pieces of paper, NumPy software is used to transform words into bits or arrays of bits because computers don't understand language.

For our training data, we're building a 3D array of all words, which is essentially a one-hot encoding of all letters and unique characters.

Later, we'll feed these X features to our model, which will have an output layer with the same node number as the input node.

IV. RESULTS AND ANALYSIS

In this section, we will see the test results of the model which is tested against several test cases. The test results help in drawing the conclusion of it and also discuss the improvements need to be done to enhance good accuracy than earlier. This even helps to focus on the future scope of this project to come in existence.

4.1 Performance Analysis:

The experimental result shows that the model has achieved 58% accuracy. The accuracy graph and model loss graph are shown below:

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

4.2 Model Accuracy:

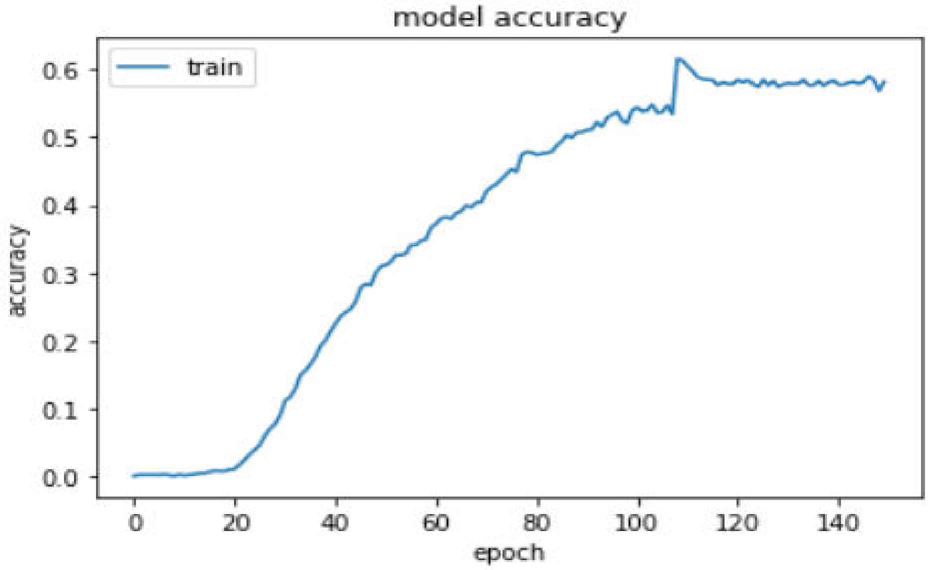


Figure 2: Model Accuracy Graph

Model Loss:

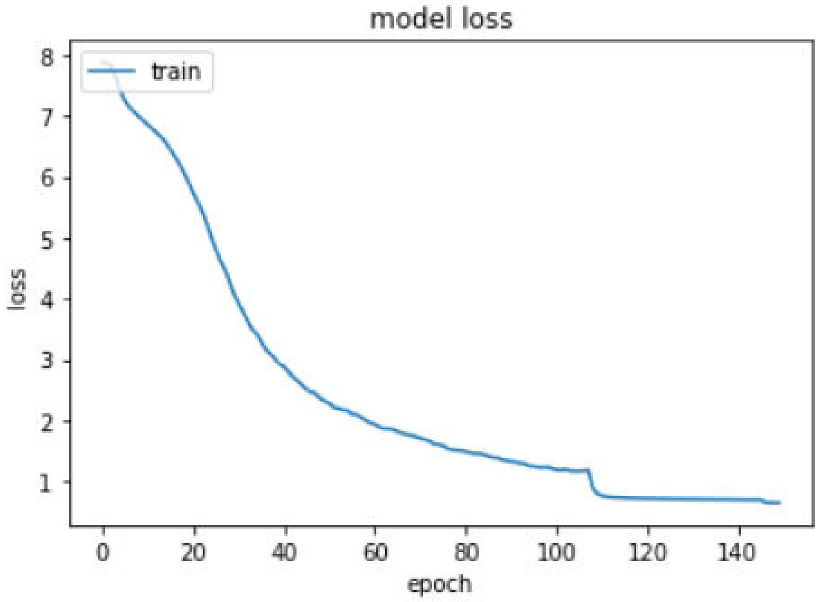


Figure 3: Model Loss Graph

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

Final Result:

The below is the Output Screenshot:

```
Enter your line: collection of textile
samples
Enter your line: what a strenuous
career
Enter your line: what
make
Enter your line: he looked over
alarm
Enter your line: He was still hurriedly thinking all this
longer
Enter your line: at the dull
weather
Enter your line: The first thing he
found
Enter your line: So why did his sister not
rung
Enter your line: Throughout all this
longer
Enter your line: It really now seemed very
entertaining
Enter your line: He rushed up to them with his arms
outstretched
Enter your line: The three gentlemen stepped out of their
misfortune
Enter your line: Then the door of the bedroom
quaking
Enter your line: leave my
child
Enter your line: stop the script
Ending The Program.....
```

Figure 4: Final output

The sentence/word is passed as the input, the predicted word is obtained as the Output which is the reflection of the trained knowledge.

If the model achieves 100% accuracy, then it is treated as overfitting. When it comes to testing, the accuracy will drop. We should be aware and remember the output of the model will be the probability that the passed input words/sentence is matched with the already trained words/sentences. With the knowledge (we have discussed this at the end of the training stage), the model comes to predict the next word of the passed input sentence/word. In order to acquire even more accuracy, we need to train the model as much as we can with new kinds of data for the model achieving more accuracy and getting into real time usage of this.

V. CONCLUSION

Robots will soon be able to read and comprehend paragraphs and stories using machine learning algorithms like RNN. To anticipate good results, we need to grasp the essential aspects of paragraphs and phrases. As new data is collected, we can train the model to reevaluate the weights. The term "paraphrase" refers to the act of taking someone else's thoughts and putting them into your own words. For our algorithms to be able to forecast greater number words when considering a single sentence and assist users in framing n number of sentences, we must recreate a passage without altering the original meaning. In general, standard RNNs and other language models are less accurate when the gap between a specific situation and the word to be expected widens. With its memory cells, an LSTM can be utilised to deal with the problem of long-term reliance. When it comes to this project, we're looking forward to using an LSTM in order to acquire a high level of accuracy because we have to forecast the user's future text that he will be thinking. Because this problem is unique, we were able to develop a 3D vector input layer and a 2D vector output layer, which we fed into an LSTM layer with hidden layers and achieved high accuracy over 150 epochs.

In this study, we show how a trained system's scalability may be expanded and its performance can be improved by employing certain strategies in the prediction and correction process..

REFERENCES

1. ParthaPratimBarman, AbhijitBoruah,"A RNN based Approach for next word prediction in Assamese Phonetic Transcription" ParthaPratim Barman et al. / Procedia Computer Science 143 (2018) 117–123.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 5, May 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1005215|

2. Prasad R," Next Word Prediction and Correction System Using Tensorflow" E-ISSN No : 2454-9916 Volume : 3 Issue : 5 May 2017.
3. Even-Zohar, Yair, and Dan Roth. "A classification approach to word prediction." Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference. Association for Computational Linguistics, 2000.
4. Hisham Al-Mubaid "A Learning-Classification Based Approach for Word Prediction", The International Arab Journal of Information Technology, Vol. 4, No. 3, July 2007.
5. Nicolas Loeff, Ali Farhadi, Ian Endres and David A. Forsyth, Unlabeled Data Improves Word Prediction, 2009 IEEE 12th International Conference on Computer Vision (ICCV).
6. Seunghak Yu, Nilesh Kulkarni, Haejun Lee, Jihie Kim," On-Device Neural Language Model based Word Prediction, Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations, pages 128–131.
7. Disha Shree Gupta, “Fundamentals of Deep Learning – Introduction to Recurrent Neural Networks”, 2017. Available:https://www.analyticsvidhya.com/blog/2017/12/introductio n-to-recurrent-neural-networks/
8. J. Yang, H. Wang and K. Guo, "Natural Language Word Prediction Model Based on Multi-Window Convolution and Residual Network," in IEEE Access, vol. 8, pp. 188036-188043, 2020, doi: 10.1109/ACCESS.2020.3031200.
9. SitiSyakirahSazali, Zainab Abu Bakar and JafreezalJaafar, “Word Prediction Algorithm in Resolving Ambiguity in Malay Text”, International Conference on Computing for Sustainable Global Development (INDIACom),2016.
10. K. Terada and Y. Watanobe, "Code Completion for Programming Education based on Recurrent Neural Network," 2019 IEEE 11th International Workshop on Computational Intelligence and Applications (IWCIA), Hiroshima, Japan, 2019, pp. 109-114, doi: 10.1109/IWCIA47330.2019.8955090.
11. Y. Ajioka and Y. Anzai, "Prediction of next alphabets and words of four sentences by adaptive junction," IJCNN-91-Seattle International Joint Conference on Neural Networks, Seattle, WA, USA, 1991, pp. 897 vol.2-, doi: 10.1109/IJCNN.1991.155477