International Journal of Computer Applications Technology and Research
Volume 12–Issue 05, 21-27, 2023, ISSN:-2319–8656
DOI:10.7753/IJCATR1205.1005

Next Word Prediction in Bodhi Language Using LSTM-based Approach

Ankush Kumar
Department of Computer Science and Engineering
School of Engineering and Technology
Sharda University
Greater Noida, India

Pushpendra Kumar Mishra
Department of Computer Science and Engineering
School of Engineering and Technology
Sharda University
Greater Noida, India

Tsering Namgail
Department of Computer Science and Engineering
School of Engineering and Technology
Sharda University
Greater Noida, India

Sandeep Kumar
Department of Computer Science and Engineering
School of Engineering and Technology
Sharda University
Greater Noida, India

Abstract: Bodhi language is one of the rare languages which is still spoken in the Leh neighborhood, Ladakh and many Tibetan regions. There is not much linguistic research done in this language. Even google translate does not work on this language. There are various types of other linguistic researches and model available on language like English and some other regional languages like Hindi, Bangla, Ukrainian etc. But there are almost negligible research and models available on Bodhi Language.

In this paper, we proposed a Language Modelling Technique using Long Short Term Memory network (LSTM) which is based on Recurrent Neural Network (RNN), using this machine learning technique we have made a model to predict the next word in bodhi language, when the user will input anything, the model will predict the next word according to the previous word(s). This model is already made for the English language but we are making the model or basically programing the model to predict the next word in the Ladakhi language which is also called as Bodhi language. This language is more complex than English language. We have tried to make the model as accurate as possible while predicting the next word in Ladakhi language. To prepare the model we have collected dataset as a large collection of Bodhi words. In this model, we have trained the model in 500 iterations (Epochs).

we used the TensorFlow, keras, dictionaries, pandas, NumPy packages. For the coding purpose we used the platform called Google Colab which is provided by google for machine learning enthusiasts.

Keywords: NLP, Next word prediction, RNN, LSTM, machine learning, deep learning.

1. INTRODUCTION

We are living in the word of machine learning in which we see different machine learning algorithms for different kind of automated works and services. Machines which work on themselves without any human touch and effort. And automatically improve themselves through experiences of previous and current work. Machines can easily recognize the patterns in between the words and things which humans cannot recognize. We very well know that. In traditional programming, the solution is the output, while the inputs are rules and data.

Whether we're merely sending messages or browsing the internet, we usually text on gadgets like our phones and desktops. While doing such things, we can notice that it proposes the next word based on what we are typing. This function, called next word prediction, foresees the word that could appear after one of our messages. It saves us time and enables us to write more swiftly and effectively. The next word prediction (NWP) problem is a significant one in the field of natural language processing. Another name for it is language modelling, and text mining is used in this case.

In this essay, we discuss the Bodhi language, an uncommon tongue that is only spoken in the Ladakh area. It is often written in Tibetan Script, and compared to most other Tibetan dialects, its sound is considerably more similar to the classical Tibetan language. It also goes by the name Bhoti language. Many of the prefix, suffix, and head characters that are silent in many other Tibetic languages, particularly Central Tibetan, are spoken by Ladakhis. The Ladakhi alphabets are shown in figure n. 1 below.

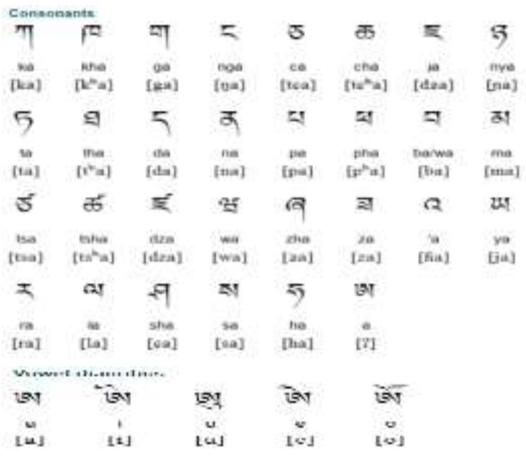

Figure 1. Alphabets in Ladakhi Script

Even Google Interpret does not have the ability to translate the Bodhi language, which is still used in Tibet and the surrounding Nepal. We are the first to attempt next word prediction in the Bodhi language, taking a step towards the strategy of preserving this extremely uncommon language. The programmers who are now working on this LSTM model are aware of how well it predicts the following word.

A variant of the recurrent neural network (RNN) architecture is long short-term memory (LSTM). However, because there is no word spacing in Bodhi, the LSTM model does not perform well when applied to this language. As a result, we must provide word space. The developers made their decision to employ LSTM because they thought it may help people remember important terms for a longer period of time. The goal of creating this model is to properly predict the next word based on input in the very language of Bodhi. In this study, we employed a variety of input changes to train the system to recognize patterns and produce correct predictions.

2. LITERATURE REVIEW

One of the current hot areas in natural language processing research is next word prediction. On this subject, numerous study papers have been published. These papers serve as the sources for our study paper.

In this paper the author suggests a method for Ukrainian language next-word prediction using a neural network-based language model.[1] To improve performance, the authors examine several model types and data pretreatment methods. They also suggest a modified version of the model that would take contextual information into account and assess the model's efficacy. The suggested method is very accurate and effective for a range of natural language processing jobs. The study advances Ukrainian, a language with limited resources, in terms of natural language processing. The quantity of the dataset utilized for assessment and the breadth of the suggested technique are two limitations of the article, though.

The use of recurrent neural networks (RNNs) for word prediction is examined in this research.[2]The authors talk on the value of next word prediction in a variety of contexts, including text completion, speech recognition, and machine translation. Following a brief introduction to RNNs and their design, they suggest a model for next word prediction that makes use of a Long Short-Term Memory (LSTM) network. On a dataset of text from diverse sources, the suggested model is assessed and contrasted with other models already in use. The findings demonstrate that, in terms of accuracy and efficiency, the suggested LSTM-based model performs better than alternative models.

The use of pre-training methods for federated text models in the context of next word prediction is explored in this work.[3]The authors provide a system to enhance next word prediction problems that blends federated learning with pre-training techniques like BERT and GPT-2. On a sizable dataset, they test their suggested framework, and they compare the outcomes to those of other cutting-edge models. In comparison to current models, the suggested framework, according to the authors, offers more accuracy and scalability.

In their study from 2020, Moghar and Hamiche [4] investigate the use of LSTM recurrent neural networks for stock market forecasting. To forecast stock values over a given time period, the authors used stock data from a significant American firm using the LSTM model. The outcomes demonstrated that the LSTM model was highly accurate in predicting stock values. The work delivers a significant addition to the field of predictive analytics and offers insights into the possible application of machine learning algorithms for financial forecasting.

The use of Long Short-Term Memory (LSTM) neural networks for the problem of next word prediction is discussed in the study. The LSTM model's design and training procedure are described by the authors, who also assess the model's performance using a dataset of text sequences.[5] In comparison to conventional language models, the experimental findings demonstrate that the LSTM model achieves excellent accuracy in predicting the following word in a given sequence. Additionally, the authors offer possible uses for the LSTM model in natural language processing tasks as speech recognition, machine translation, and text creation.

The study on COVID-19 time series forecasting in Canada using Long Short-Term Memory (LSTM) networks is presented in the publication.[6] The authors assess three LSTM models using a dataset of daily COVID-19 cases in Canada and compare their performance using various input settings. With the best model reaching a mean absolute percentage error of 5.78%, the results show that the LSTM models are successful in projecting the COVID-19 transmission. According to the study, LSTM networks can be an effective tool for forecasting COVID-19's future spread and guiding public health policy in Canada.

The study on utilizing Long Short-Term Memory (LSTM) neural networks for projecting an object's future location is presented in the publication "Next position prediction using LSTM neural networks."[7] The study focuses on using a particular dataset that contains information on the position and speed of a moving item. The LSTM neural network model is applied to this dataset by the authors, who then assess its performance using several criteria. They contrast the outcomes with those from several forecasting methods, such as Linear Regression and Random Forest. According to the study's findings, The LSTM neural network model performs better than the other models in terms of prediction accuracy, indicating its potential for usage in related applications in the future.

The exploration by Fang, Chen, and Xue [8] gives a survey of the literature on spatio-temporal sequence prediction algorithms based on recurrent neural networks (RNNs). Before discussing several RNN models and their variations that have been suggested for this job, the authors give an outline of the key ideas and difficulties in spatiotemporal data prediction. They emphasize the benefits and drawbacks of each model and shed light on the direction that this field of study is now taking. Overall, the publication is a valuable tool for scientists working on RNN-based models for spatio-temporal sequence prediction.

To predict cluster CPU consumption, Nashold and Krishnan (2020) suggested using Long Short-Term Memory (LSTM) and Seasonal Autoregressive Integrated Moving Average (SARIMA) models.[9] They tested the models using actual data and discovered that, in terms of accuracy and root mean squared error, the LSTM model performed better than the SARIMA model. The study shows how machine learning approaches might be used to enhance resource management in computer clusters. The study might benefit from further validation using more datasets as it is constrained by the use of a single dataset.

Long Short-Term Memory (LSTM) neural networks are suggested as a method in the study "Real-time driver maneuver prediction using LSTM" for the prediction of driver maneuvers in real-time.[10] The accuracy of the model that the authors developed for predicting four different driving maneuvers—changing lanes, turning left, turning right, and stopping—was 87.5% when tested on data from a driving simulator.Additionally, they evaluated the effectiveness of their model against other cutting-edge techniques. The suggested approach may be used in traffic management, driver aid systems, and driverless cars.

A paradigm for context-based text production that makes use of Long Short-Term Memory (LSTM) networks was put out by Santhanam in 2020.[11] To produce writing that is cohesive and pertinent, the model takes into account the context of the words before them. The suggested model's architecture and training procedure are described in detail, and the study assesses the model's performance using a number of measures. The outcomes demonstrated that the suggested model performs better at producing pertinent and coherent content than conventional language models. According to the article, the suggested model may find use in a number of natural languages processing tasks, including chatbots, conversation systems, and machine translation.

The approach for forecasting product quality in the research combines time-dimensional K-means with state transition-LSTM [12] networks, multiple models, and dual sampling periods. The authors do trials using real-world datasets to show that their strategy performs better in terms of prediction accuracy than the competition. They also shed light on the significance of including both past and present data in prediction processes. In order to increase product quality and lower production costs, the study demonstrates the possibility of applying machine learning approaches for quality prediction in industrial settings.

The study describes an intelligent, autonomous street lighting system that saves energy by using weather forecast information. The Long Short-Term Memory (LSTM) algorithm [13] serves as the system's foundation for forecasting weather conditions and altering illumination settings accordingly. The results of the trials the authors ran to compare the suggested system's energy usage with a conventional street lighting system revealed a considerable reduction in energy consumption. The potential of data-driven methodologies for developing sustainable energy solutions is highlighted in the article.

A thorough analysis and assessment of text prediction and entertainment systems are provided in the work by Hamarashid, Saeed, and Rashid.[14] The authors look at several methods for text prediction, such as rule-based and machine learning-based ones, and rate their precision and efficacy. They also talk about how chatbots and video games that employ text prediction are used for fun. The study emphasizes the significance of enhancing these systems' performance through the application of cutting-edge algorithms and methodologies and offers insights into prospective future research topics.

The study suggests a unique method for employing LSTM recurrent neural networks to forecast numerous illnesses. The authors used a collection of patient medical information from a variety of conditions to conduct tests.[15] In terms of accuracy, sensitivity, and specificity, the suggested model produces encouraging results. The work emphasizes the opportunity for multi-disease prediction using deep learning models, which has the potential to greatly enhance healthcare services.

The research examines the cloud sentiment analysis accuracy of three recurrent neural network models: RNN, LSTM, and GRU. The data was pre-processed by the authors [16] using methods including tokenization, stemming, and stop-word removal from internet evaluations of cloud services. The three models were then trained, and their accuracy was assessed using measures including precision, recall, and F1-score. With an F1-score of 0.8625, the LSTM model beat the other two models. The GRU model came in second with an F1-score of 0.8573, while the RNN model came in third with an F1-score of 0.8248. For cloud sentiment analysis, the authors advise utilizing LSTM.

The Long Short-Term Memory (LSTM) algorithm and local weather predictions are combined in the paper's innovative technique for forecasting [17] hourly solar irradiance. The LSTM model is evaluated on a dataset from a solar power facility in South Korea after being trained using non-local meteorological data. Results reveal that the suggested technique performs better in terms of accuracy than other current models and can forecast the hourly solar irradiance for the following day with an average error rate of 9.73%. According to the study, energy management systems may employ the suggested technique for improved planning and management of solar power generation.

The usefulness of machine learning and deep learning approaches for forecasting stock prices is examined in the study "Stock price prediction using machine learning and LSTM-based deep learning models".[18] The study forecasts stock values using historical data utilizing LSTM-based deep learning models and a variety of machine learning methods, such as Random Forest and Support Vector Regression. The article comes to the conclusion that the LSTM-based model performed better than the conventional machine learning methods and might be a useful stock price prediction tool. The study's overall findings emphasize the promise of deep learning methods for stock market forecasting.

The EEMD-GA-LSTM approach using large scaled wind history data is the basis for the novel framework for short-term wind speed prediction proposed in this research.[19] The decomposition of the wind speed time series into intrinsic mode functions using the proposed method's ensemble empirical mode decomposition (EEMD) algorithm is followed by genetic algorithm (GA) optimization to choose the most pertinent features. Finally, wind speed prediction is performed using a long short-term memory (LSTM) model. The EEMD-GA-LSTM technique performs better than numerous other approaches in terms of prediction accuracy and resilience when the suggested method is evaluated on actual wind speed data, making it a viable method for short-term wind speed prediction.

The Long Short-Term Memory (LSTM) and bi-directional LSTM (BLSTM) models are highlighted in this paper's evaluation of the literature on the application of deep learning techniques for stock price prediction.[20] The authors describe other research that have utilized these models for stock price prediction and go through the benefits of using them over more conventional approaches.

They also list some of the drawbacks of this strategy, such as the requirement for a lot of data and the complexity of interpreting the findings. Overall, the study offers a valuable summary of the current status of LSTM and BLSTM-based deep learning-based stock price prediction.

3. METHODOLOGY

3.1 Dataset

The dataset was compiled from around 5000 Bodhi words which were from various Ladakhi articles, newspapers, and dictionaries. Sentences made up of words are then combined to serve as the input for training and testing. These sentences have a word count of five or more. Many brief sentences with sequences of less than six words can be found in the dataset. Fig. 2 shows the snap of the dataset.

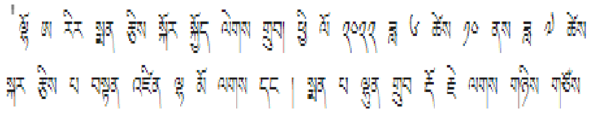

Figure 2. Snap of the dataset

3.2 Methodology

We used a LSTM based approach to build our model in which we build a sequential model with 4 layers in which two LSTM layers and two Dense layers. The input size of the model is 3 and output is 1 while the sequence length is 10. We trained our model in batch size of 64 and in 500 epochs. To maintain and update the state of memory cells, the LSTM model filters data through the gate structure. Its door structure consists of input, output, and forgotten gates.[18] Three sigmoid layers and one tanh layer make up each memory cell. The LSTM memory cells' structure is shown in Fig. 3.

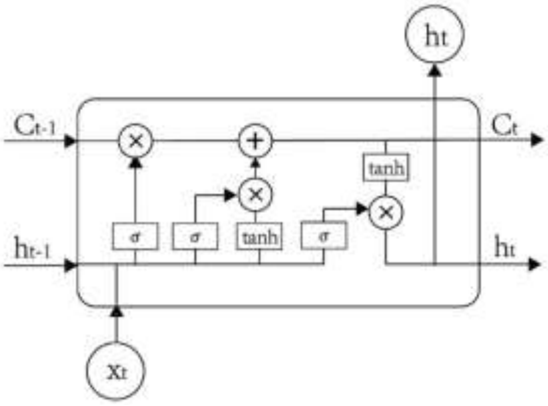

Figure 3. Basic Architecture of LSTM

Which cell state information is eliminated from the model by the forgotten gate in the LSTM unit. As seen in Fig. 1, the memory cell takes as inputs the external information xt of the present instant and the output ht-1 of the previous moment, which it then combines into a long vector [ht-1, xt] by transformation to create

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f),$$

where Wf and bf stand for the forgetting gate's weight matrix and bias, respectively, and is the sigmoid function. The primary purpose of the forgotten gate is to keep track of how much of the current cell state Ct is reserved for the prior cell state Ct-1. Based on ht-1 and xt, the gate will output a value between 0 and 1, with 1 denoting total reserve and 0 denoting complete discard.

In figure 3, By allowing only a few linear interactions, the cell state of LSTM helps the information to flow through the units without being altered. Each unit has input, output and a forget gate. Each unit with the help of input, output and forget gate can add or remove information. Forget gate decides which information is to be forgotten and which is not to be forgotten. Forget gate uses sigmoid function for it. While the input gate controls the flow of information. Input gate uses pointwise multiplication of Sigmoid and tanh function for it respectively. While the output gate decides which information to passed to next state.

Figure. 4 shows the most basic outline of our model in which 3 words as input are required while it gives the next word as output.

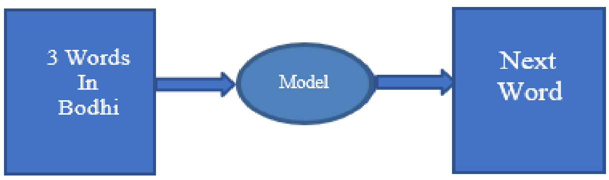

Figure 4. basic outline of the model

For the preparation of such a model, we have to work on the dataset using various operations and functions provided by various ML related packages in python such as numpy, Tensorflow, Pandas, Matplotlib, Tokenizer etc.

To prepare the model we divided the model into modules which solves the sub problems. The dataflow between these modules of sub- tasks is given in Figure. 5.

The figure describes how data flows from various sub tasks which performs some operations on data including cleaning and preprocessing of data to training the model and after that the prediction of the next word. It only shows the sub tasks for preparing the model and the data flow through it. It does not show how many layers are in the model and how and which operations are performed in the subtask.

Figure 5. Applied Basic Method

All the processes or operations performed on data before feature engineering comes under the data-preprocessing in which cleaning the dataset, removing outliers and converting the data into machine understandable language or in binary digits comes. Before starting the procedure, the dataset must be cleaned. The words in the dataset are then divided into several groups.

The iterator is used to parse the input file and collect distinct words. With the help of tokenizer function in preprocessing library of tensorflow package tokens were generated for sequential text. This demonstrates that words are challenging for machine learning neural networks to process, making it essential to map them to indices, which are straightforward for neural networks to recognize.

The process starts with the sequencing of the given words after the sequencing the feature engineering is done and input and output features were created. For prediction, we used 3 words to predict the next words for which the input was the np array with 4 sequence elements and output was the last element of the np array. We used a Sequential model with 4 layers. Our model was fairly accurate. We used our model to predict in both languages in English as well as bodhi language. Model was fairly accurate in English language but its accuracy decreased in Bodhi language. Yet we tried to raise its accuracy by separating the words and then tokenization of these words. In English language, we trained our model for only 7 Epochs in last epoch loss was equal to 4.4253 and accuracy was around 70%. Which is shown in below figure.

```
Epoch 7/7
1958/1958 [==============================] - ETA: 0s - loss: 4.4253
Epoch 7: loss improved from 4.62284 to 4.42526, saving model to next_words.h5
1958/1958 [==============================] - 30s 15ms/step - loss: 4.4253
```

Figure 6. Epochs for next word prediction for English language

In Bodhi language we trained our model for 500 iterations and loss was around 0.00022 and accuracy was 50%. Which is show in the below figure.

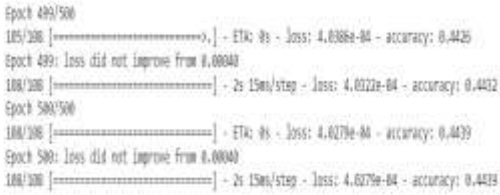

Figure 7. Epochs for next word prediction model for Bodhi Language

Table 1. Comparison of accuracy between English and Bodhi language accuracy.

	Epochs	Loss	Accuracy (in %)
English	7	4.42	70
Bodhi	500	0.0003	50

As this language is very hard for preprocessing and training the model, yet we managed to train our model, and test it manually. The loss of our model kept on decreasing in our model. It was very hard to collect dataset for training our model and in future as dataset increases our model will improve in accuracy as well as performance. We prepared a separate function for preparation of user input. This function takes input from user tokenize it and preprocessing happen which converts the input text into tokens with indexing of serial numbers. Then the input is given to model. Model works on input and gives output as a word.

4. RESULT

While training our model the loss kept on decreasing. Loss to Epochs graph is given below which visualizes how the loss is behaving in each epoch. The figure given below visualizes the trend of loss during training (figure 8 shows). Which is also shown using table n. 2.

Figure 8. Training loss

Accuracy trends of our model throughout all training epochs are visualized by the figure n. 9 given below also the same is shown using the table n. 3.

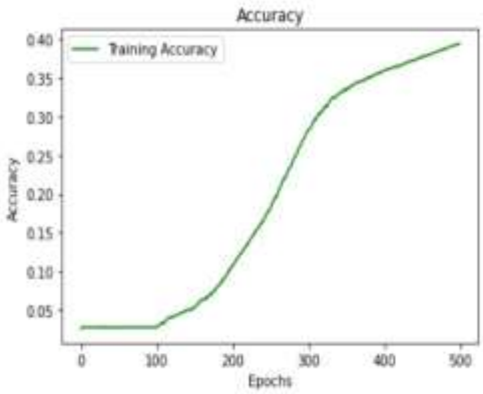

Figure 9. Accuracy

Table 2. Training loss throughout training of model

Loss	Epochs
0.00044	0-100
0.00042	100–150
0.00040	150-200
0.00036	200-250
0.00032	250-290
0.00025	290-300
0.00022	300-500

Table 3. Accuracy throughout training of model

Accuracy	Epochs
0.13	0-200
0.28	200-300
0.38	300-400
0.44	400-500

We created a function for preparing input and feeding it to our model. With the help of the tokenization our input text was sequenced. our model processes the input and gives one and the most probable word as output.
Now our model can predict the next word in Bodhi language. Which is the language of Ladakh and surrounding regions Nepal and Tibet.

We used 3 words as input and accuracy was fairly high in comparison to 1 word as input which is 50%.
A snap of the output of the model is given below in figure n. 10. including the parts of

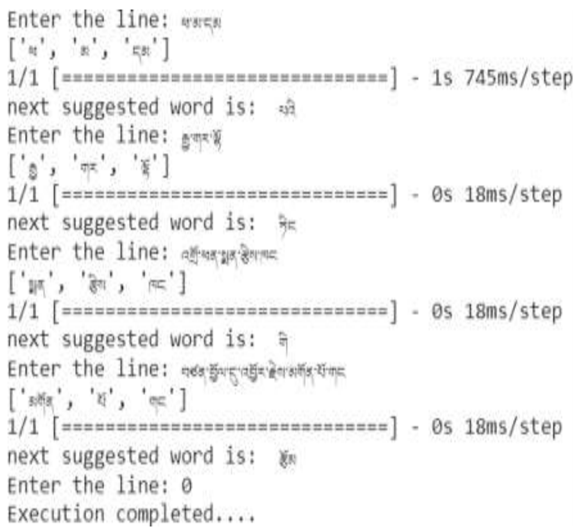

Figure n. 10: Snap of Output

5. CONCLUSION

Next word prediction is one of the most researched NLP fields because it is about finding text. We have used an LSTM model which is trained in 500 iterations. Our work is the first to be done on a rare language like Bodhi. From the result, it can be said that the accuracy is sufficiently high. This model can be used to predict the next word from the target's input. This model works on a language which even google translate does not provide a service for, which is a feat in itself. First our model takes words as input for which we have decided to take 3 words as minimum word limit so our model takes minimum 3 words as input and assign tokens to these words using tokenizer function which is already available after tokenizing model finds the next word based on these three words and gives the most probable word as input. For testing of our model, we had to make a module separately so that model can tokenize the input words and take those words as input. Because all the work is going on in numerical terms and model tries to find the underlying patterns among these numerical tokenized sequences which are given for the words while training.

6. FUTURE SCOPE

As the dataset was pretty hard to collect, hence in the future we can gather more data so that our model can be trained thoroughly. And its accuracy can be increased. There is a need to research the proper method for the preprocessing of dataset in Bodhi language so that accuracy can be increased. Future research may be done to improve next word prediction models' performance for Bodhi and other low-resource languages. The method used in this work may also be applied to other language models that call for next word prediction, which will facilitate the creation of natural language processing tools that are more precise and efficient. The results of this study demonstrate the potential of deep learning in natural language processing and its capacity to promote the growth of underdeveloped languages.

7. REFERENCES

[1] Shakhovska, K., Dumyn, I., Kryvinska, N., & Kagita, M. K. (2021). An Approach for a Next-Word Prediction for Ukrainian Language. Wireless Communications and Mobile Computing, 2021, 1-9.

[2] Ambulgekar, S., Malewadikar, S., Garande, R., & Joshi, B. (2021). Next Words Prediction Using Recurrent NeuralNetworks. In ITM Web of Conferences (Vol. 40, p. 03034). EDP Sciences.

[3] Stremmel, J., & Singh, A. (2021). Pretraining federated text models for next word prediction. In Advances in Information and Communication: Proceedings of the 2021 Future of Information and Communication Conference (FICC), Volume 2 (pp. 477-488). Springer International Publishing.

[4] Moghar, A., & Hamiche, M. (2020). Stock market prediction using LSTM recurrent neural network. Procedia Computer Science, 170, 1168-1173.

[5] Afika, R., Suprih, W., Atikah, D. A., & Fadlan, B. H. (2022). Next word prediction using LSTM. Journal of Information Technology and Its Utilization, 5(1), 10-13.

[6] Chimmula, V. K. R., & Zhang, L. (2020). Time series forecasting of COVID-19 transmission in Canada using LSTM networks. Chaos, Solitons & Fractals, 135, 109864.

[7] Violos, J., Tsanakas, S., Androutsopoulou, M., Palaiokrassas, G., & Varvarigou, T. (2020, September). Next position prediction using LSTM neural networks. In 11th Hellenic Conference on Artificial Intelligence (pp. 232-240).

[8] Fang, W., Chen, Y., & Xue, Q. (2021). Survey on research of RNN-based spatio-temporal sequence prediction algorithms. Journal on Big Data, 3(3), 97.

[9] Nashold, L., & Krishnan, R. (2020). Using LSTM and SARIMA models to forecast cluster CPU usage. arXiv preprint arXiv:2007.08092.

[10] Khairdoost, N., Shirpour, M., Bauer, M. A., & Beauchemin, S. S. (2020). Real-time driver maneuver prediction using LSTM. IEEE Transactions on Intelligent Vehicles, 5(4), 714-724.

[11] Santhanam, S. (2020). Context based text-generation using lstm networks. arXiv preprint arXiv:2005.00048.

[12] Shi, X., Li, Y., Yang, Y., Sun, B., & Qi, F. (2021). Multi-models and dual-sampling periods quality prediction with time-dimensional K-means and state transition-LSTM network. Information Sciences, 580, 917-933.

[13] Tukymbekov, D., Saymbetov, A., Nurgaliyev, M., Kuttybay, N., Dosymbetova, G., & Svanbayev, Y. (2021). Intelligent autonomous street lighting system based on weather forecast using LSTM. Energy, 231, 120902.

[14] Hamarashid, H. K., Saeed, S. A., & Rashid, T. A. (2022). A comprehensive review and evaluation on text predictive and entertainment systems. Soft Computing, 1-22.

[15] Men, L., Ilk, N., Tang, X., & Liu, Y. (2021). Multi-disease prediction using LSTM recurrent neural networks. Expert Systems with Applications, 177, 114905.

[16] Raza, M. R., Hussain, W., & Merigó, J. M. (2021, October). Cloud sentiment accuracy comparison using RNN, LSTM and GRU. In 2021 Innovations in intelligent systems and applications conference (ASYU) (pp. 1-5). IEEE.

[17] Jeon, B. K., & Kim, E. J. (2020). Next-day prediction of hourly solar irradiance using local weather forecasts and LSTM trained with non-local data. Energies, 13(20), 5258.

[18] Mehtab, S., Sen, J., & Dutta, A. (2021). Stock price prediction using machine learning and LSTM-based deep learning models. In Machine Learning and Metaheuristics Algorithms, and Applications: Second Symposium, SoMMA 2020, Chennai, India, October 14–17, 2020, Revised Selected Papers 2 (pp. 88-106). Springer Singapore.

[19] Chen, Y., Dong, Z., Wang, Y., Su, J., Han, Z., Zhou, D., ... & Bao, Y. (2021). Short-term wind speed predicting framework based on EEMD-GA-LSTM method under large scaled wind history. Energy Conversion and Management, 227, 113559.

[20] Sunny, M. A. I., Maswood, M. M. S., & Alharbi, A. G. (2020, October). Deep learning-based stock price prediction using LSTM and bi-directional LSTM model. In 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES) (pp. 87-92). IEEE.

[21] Mei, L., Hu, R., Cao, H., Liu, Y., Han, Z., Li, F., & Li, J. (2020). Realtime mobile bandwidth prediction using LSTM neural network and Bayesian fusion. Computer Networks, 182, 107515.

[22] Zhao, J., Zeng, D., Xiao, Y., Che, L., & Wang, M. (2020). User personality prediction based on topic preference and sentiment analysis using LSTM model. Pattern Recognition Letters, 138, 397-402.

[23] Mueller, A., Nicolai, G., Petrou-Zeniou, P., Talmina, N., & Linzen, T. (2020). Cross-linguistic syntactic evaluation of word prediction models. arXiv preprint arXiv:2005.00187.

[24] Hansun, S., & Young, J. C. (2021). Predicting LQ45 financial sector indices using RNN-LSTM. Journal of Big Data, 8(1), 1-13.

[25] Rauf, H. T., Lali, M. I. U., Khan, M. A., Kadry, S., Alolaiyan, H., Razaq, A., & Irfan, R. (2021). Time series forecasting of COVID-19 transmission in Asia Pacific countries using deep neural networks. Personal and Ubiquitous Computing, 1-18.

[26] Kwon, B. S., Park, R. J., & Song, K. B. (2020). Short-term load forecasting based on deep neural networks using LSTM layer. Journal of Electrical Engineering & Technology, 15, 150