Predicting the next word using Long short-term modeling(LSTM)

Subbarao Sanka
subbaraosanka@my.unt.edu
Masters in computer science at UNT

Srilatha Kudumula
Srilathakudumula@my.unt.edu
Masters in computer science at UNT

Rohit Bodepudi
bodepudirohit@my.unt.edu
Master in computer science at UNT

Dileep Reddy Tenali
dileepreddytenali@my.unt.edu
Masters in computer science at UNT

Manasa Kalvacherala
manasakalvacherla@my.unt.edu
Masters in computer science at UNT

***Abstract*— This report discusses the use of RNN and LSTM to predict the next word while typing anything by utilizing a variety of preprocessing techniques, models, and online deployments more accurately.**

Predicting the next word is also known as language processing which is a branch of Natural Language Processing (NLP). A tree-based generative language model has been applied for rating documents and portions. Nodes in the tree represent various document elements, such as titles, paragraphs, and sections. The document tree has a clear linguistic model at each node. A leaf node's language model can be inferred directly from the text of the document portion that corresponds to the node. Using a linear interpolation between the different young nodes, inner nodes inside the tree can be predicted. The study also explains how this model might provide satisfactory answers to several typical structural issues.

Keywords—Language Modeling (LM), Natural Language Processing(NLP), Long short-term Memory(LSTM),

I. Introduction

Deep learning itself is part of machine learning where it uses the different layers of neural networks that decide classifications and predictions.

Almost, we use gadgets every day in our life, then we use typing either for searching anything on the gadget or browsing the internet using keyboards or other alternatives. During those activities, we see the next word while typing, so, here comes the next word prediction while typing. Next word prediction is the arena of natural language processing.

Predicting the next word using LSTM (Long short-term memory) is a variant of RNN. The reason to use this model is it has a longer memory for predicting the next word.

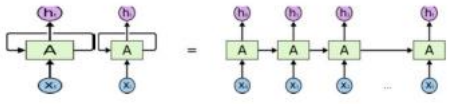


Fig1: Recurrent Neural Network

In the above diagram, the RNN, A takes the x_i as input and produces the h_i as output. The information pass from one network to another in a loop. The RNN is a copy of multiple standard neural networks, which passes the message individually to a successor. Recurrent neural networks are state-of-the-art algorithms for successive information. We also use the LSTM (Long short-term memory) because of its hidden layer's state. Using it we can add/subtract to preserve information for a long time. The LSTM cell will take a new value at each step.

II. Architecture and Methodology

As a first step, it will get the user's input and save the user's keystrokes. Next, it will use the trained model to predict the next word. RNNs are used for Grammarly and spelling correction on the basis of the training model.

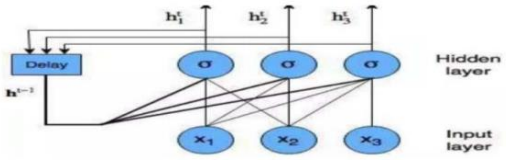


Fig 2: Recurrent neural network with hubs.

In the above figure, the three enter hubs will be fed into the hidden layer with sigmoid activations and thickly associate with the neural system. The hidden layers are passed across the delay blocks for contributions of h_i.

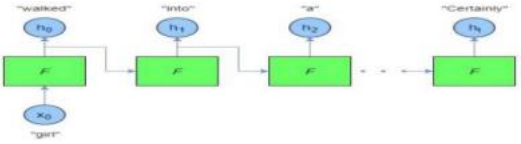


Fig3: One-to-many configurations of RNN

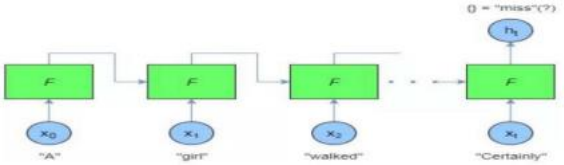


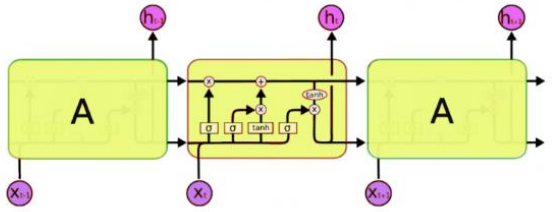
Fig4: many-to-many configurations of RNN

Above figures 3 & 4 show the one-to-many and many-to-many configurations of RNN.

III. Architecture

LSTM is the most effective algorithm when compared to the backpropagation algorithm. LSTM uses feedback connections. It has RNN which can process entire sequences of data rather than single data points. LSTM includes both long and short-term memory. LSTM includes a cell, input gate, output gate, and forget gate. These three gates help in the flow of information from in the cell and the cell helps in remembering the time intervals

LSTM helps in making the predictions, data classifying, processing based on time series



FORGET GATE: The main intention of the forgetting gate is to remove the data that is not required for prediction, or which is given less importance. It helps in optimizing the performance

INPUT GATE: This input gate is responsible for sending the information to the cell It includes three steps

Step 1: values needed to be added to the cell using the sigmoid function

Step 2: Create a vector to store all the input values

Step 3: multiplying the values using filters

OUTPUT GATE: Based on input data given to cell output will be predicted

IV. Tools and Technology used

- Tensor flow to create large-scale neural networks.
- Python3
- AWS
- Dataset used for this project will be https://www.kaggle.com/code/carlosaguayo/predicting-the-next-word-using-lstm/data.

V. Use cases

- Search engines
- Smartphone keyboards

VI. Conclusion

With the solution we proposed, A study using deep learning methods to predict the next word based on the user input.

VII. References

[1] R. Kneser and H. Ney. Improved backing-off for ngram language modeling. In International Conference on Acoustics, Speech and Signal Processing, pages 181–184, 1995.

[2] S.F. Chen and J.T. Goodman. An empirical study of smoothing techniques for language modeling. Computer, Speech and Language, 13(4):359–393, 1999.

[3] J .Goodman. A bit of progress in language modeling. Technical Report MSR-TR-2001–72, Microsoft Research, 2001.

[4] Language Modelling: Next word Prediction by Mohd.Majid and Piyush Kumar,Project Thesis Report 2019.