# TEXT GENERATION USING LSTM AND IBM WATSON STUDIO

**TEAM MEMBERS:**

NAVANEETHAN P

PRAVEEN KUMAR NK

VISHAL PMB

## INTRODUCTION:

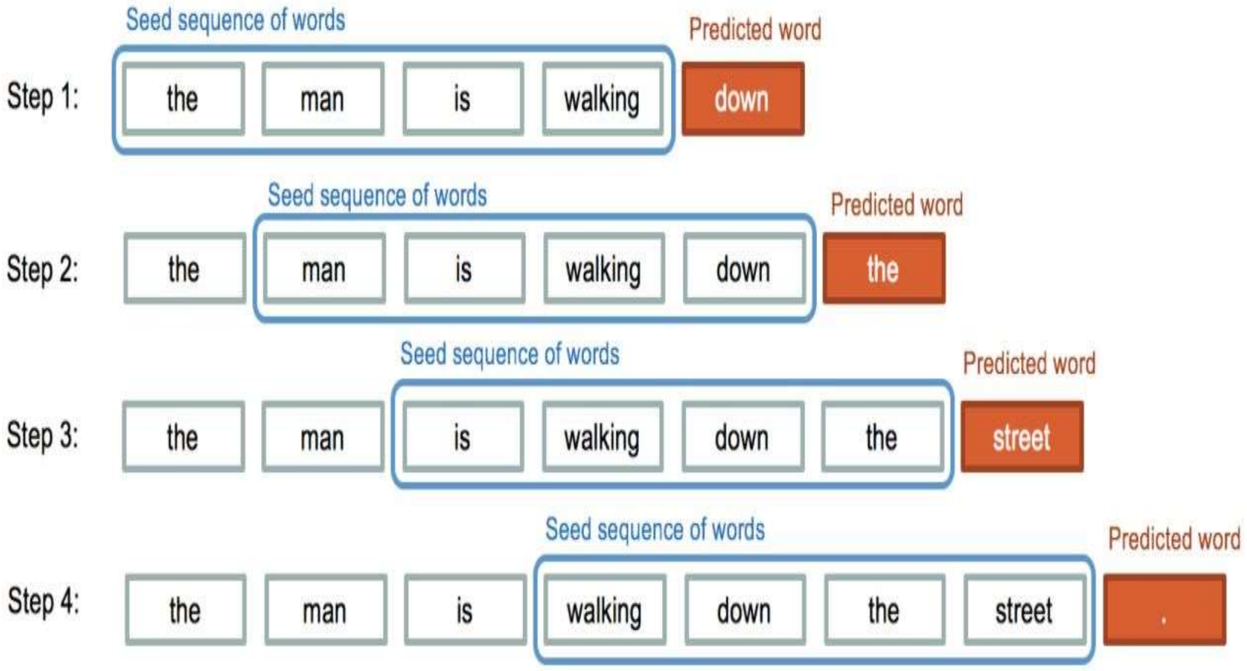
In text generation, we try to predict the next character or word of the sequence. The text data generally considered as sequence of data. For predicting data in sequence we used deep learning models like RNN or LSTM. LSTM are preferred over RNN in this because of RNN vanishing and exploding gradients problem. In text generation we have to memorize large amount of previous data. So for this purpose LSTM are preferred.

## PROBLEM STATEMENT:

One way of looking at this vanishing gradient problem is that the network is trying to propagate all the information through longer sequences. And with longer sequences, the gradients are getting diminished during BPTT. To remedy this issue, Long short term memory network (LSTM) [is introduced.

## SOLUTION:

The phrases in text are nothing but sequence of words. So, LSTM can be used to predict the next word. The neural network take sequence of words as input and output will be a matrix of probability for each word from dictionary to be next of given sequence. The model will also learn how much similarity is between each words or characters and will calculate the probability of each, Using that we will predict or generate next word or character of sequence.



## LITERATURE SURVEY:

1. Text Generation for Imbalanced Text Classification

   The problem of imbalanced data can be frequently found in the real-world data. It leads to the bias of classification models, that is, the models predict most samples as major classes which are often the negative class. In this research, text generation techniques were used to generate synthetic minority class samples to make the text dataset balanced. Two text generation methods: the text generation using Markov Chains and the text generation using Long Short-term Memory (LSTM) networks were applied and compared in the term of ability to improve the performance of imbalanced text classification. Our experimental study is based on LSTM networks classifier.

2. Word Level LSTM and Recurrent Neural Network for Automatic Text Generation

   Sequence prediction problems have been a major problem for a long time. Recurrent Neural Network (RNN) has been a good solution for sequential prediction problems. This work aims to create a generative model for text. Even though, RNN has its own limitations such as vanishing and exploding gradient descent problems, and inefficiency to keep track of long-term dependencies. To overcome these drawbacks, Long Short Term Memory (LSTM) has been a path-breaking solution to deal with sequential data and text data in particular. This paper delineates the design and working of text generation using word-level LSTM-RNN.

3. LSTM based brain-machine interface tool for text generation through eyes blinking detection

   This paper presents the development of a tool that allows people to communicate, by using only their voluntary eyes blinks. This tool provides a communication link mainly for people with motor disabilities, who cannot communicate through voice. the electroencephalographic (EEG) signal provided by the Mindwave Mobile 2 headset is the source of information to detect the voluntary blinks.

4. Towards Improved Classification Accuracy on Highly Imbalanced Text Dataset Using Deep Neural Language Models

   Data imbalance is a frequently occurring problem in classification tasks where the number of samples in one category exceeds the amount in others. Quite often, the minority class data is of great importance representing concepts of interest and is often challenging to obtain in real-life scenarios and applications. Imagine a customers’ dataset for bank loans-majority of the instances belong to non-defaulter class, only a small number of customers would be labeled as defaulters, however, the performance accuracy is more important on defaulters labels than non-defaulter in such highly imbalance datasets.

5. Improved sequence generation model for multi-label classification via CNN and initialized fully connection

   In multi-label text classification, considering the correlation between labels is an important yet challenging task due to the combination possibility in the label space increasing exponentially. In recent years, neural network models have been widely applied and gradually achieved satisfactory performance in this field. However, existing methods either not model the fully internal correlations among labels or not capture the local and global semantic information of text simultaneously, which somewhat affects the classification results finally. In this paper, we implement a novel model for multi-label classification based on sequence-to-sequence learning, in which two different neural network modules are employed, named encoder and decoder respectively.

## EXPERIMENTAL INVESTIGATIONS:

- At the beginning of the project we took nearly 60+ lakh characters as dataset which nearly estimated 160 hrs of training the dataset
- Later due to long period we changed our dataset which had 2+ lakh character only took 150 min to train.
- We did not have better graphics card to handle to use the other dataset
- It generally conforms to the line format observed in the original text of less than 80 characters before a new line.
- The characters are separated into word-like groups and most groups are actual English words (e.g. “the”, “little” and “was”), but many do not (e.g. “lott”, “tiie” and “taede”).

## HARDWARE, SOFTWARE SPECIFICATION:

**Minimum supported hardware**

**Processor -** Intel® x86 1 GHz or equivalent processor

**Memory (RAM) -** 3 GB (default install)

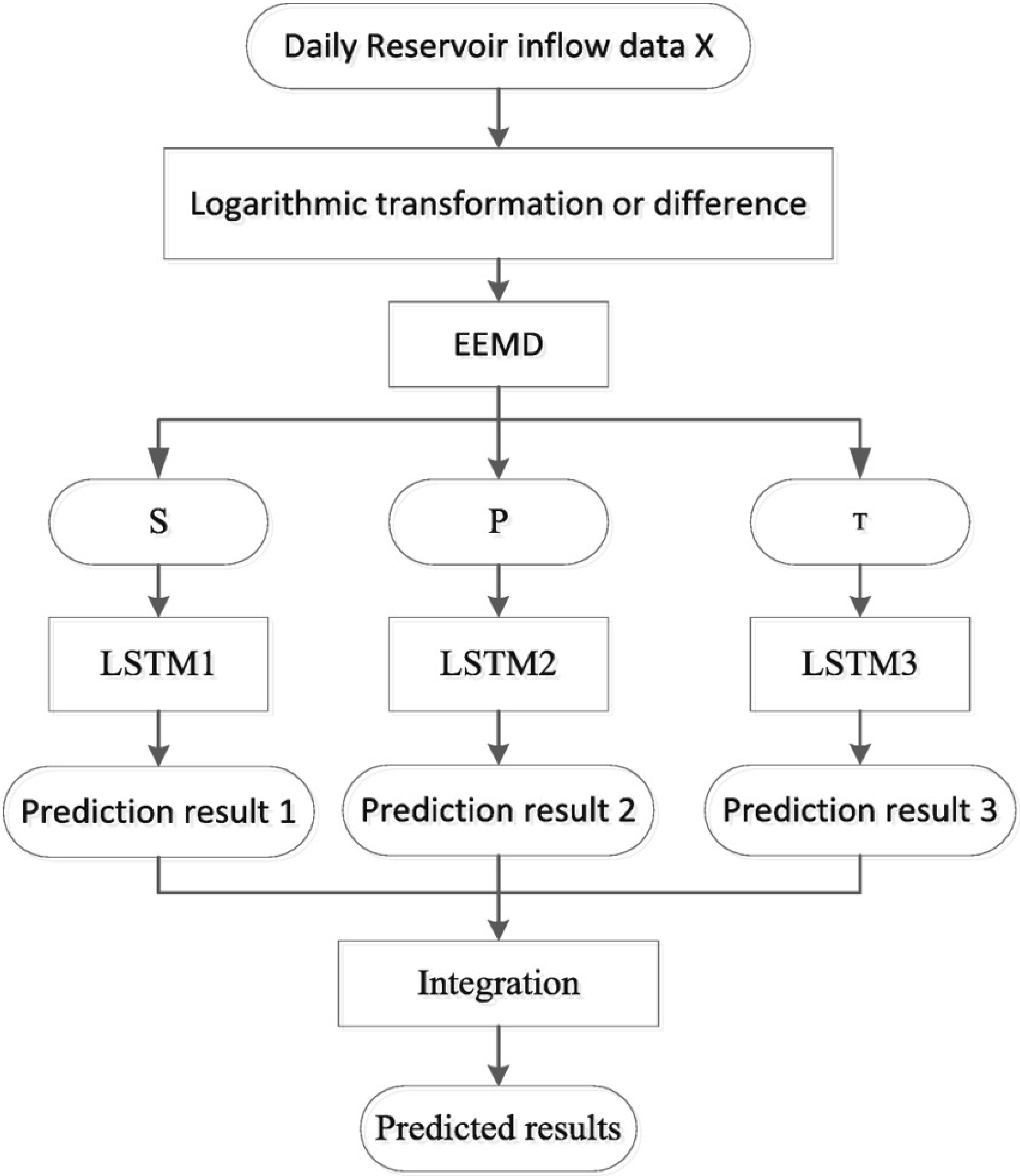
**Network** - TCP/IP

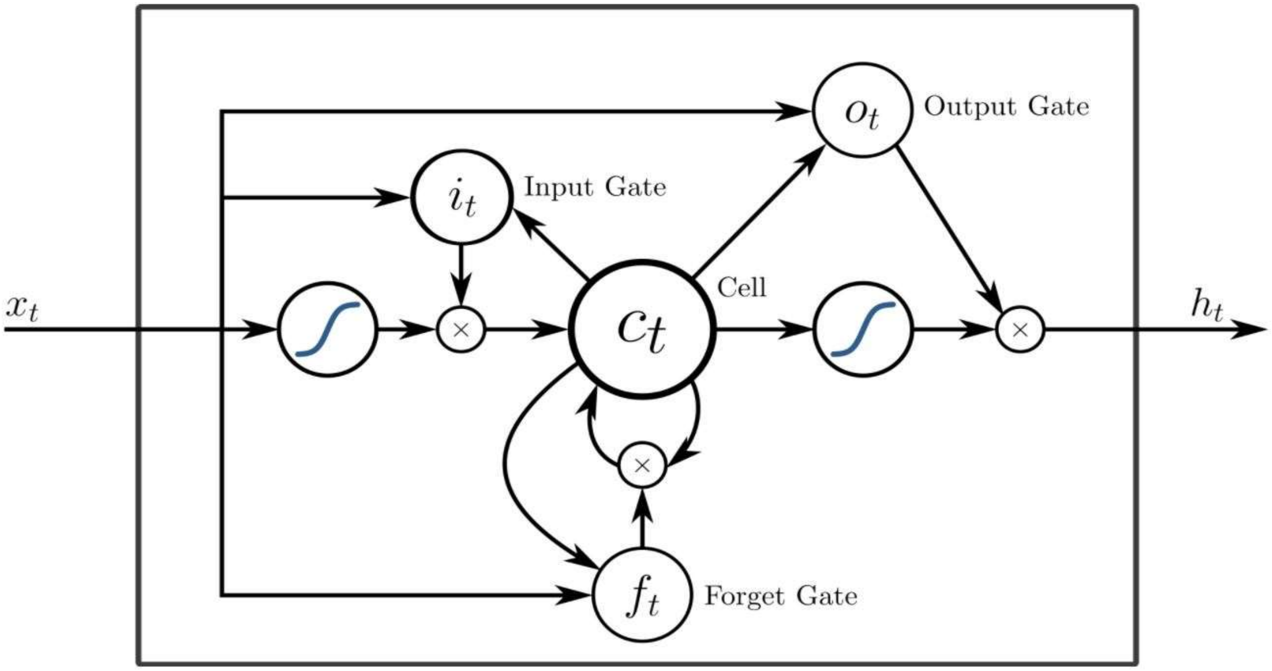
**GPU** - Nvidia Geforce cards

**Software requirements**

- Tensor flow 2.5 version
- Anaconda Navigator
- Spider

- IBM Watson studio
- IBM cloud
- **FLOWCHART:**





## CONCLUSION:

Text Generation is a type of Language Modeling problem. Language Modeling is the core problem for a number of natural language processing tasks such as speech to text, conversational system, and text summarization. A trained language model learns the likelihood of occurrence of a word based on the previous sequence of words used in the text. Language models can be operated at character level, n-gram level, sentence level or even paragraph level. In this project, we are creating a language model for generating natural language text by implementing and training state-of-the-art Recurrent Neural Network.

The fact that this character based model of the book produces output like this is very impressive. It gives you a sense of the learning capabilities of LSTM networks. Your results may vary given the stochastic nature of the algorithm or evaluation procedure, or differences in numerical precision. Consider running the example a few times and compare the average outcome

## FUTURE SCOPE:

- Predict fewer than 1,000 characters as output for a given seed.
- Remove all punctuation from the source text, and therefore from the models' vocabulary.
- Try a one hot encoded for the input sequences.
- Train the model on padded sentences rather than random sequences of characters.
- Increase the number of training epochs to 100 or many hundreds.

## BIBLIOGRAPHY:

[1]. Akkaradamrongrat, Suphamongkol; Kachamas, Pornpimon; Sinthupinyo, Sukree. In: 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE) Computer Science and Software Engineering (JCSSE), 2019 16th International Joint Conference on. :181-186 Jul, 2019;

[2]. Buddana, Harsha Vardhana Krishna Sai; Kaushik, Surampudi Sai; Manogna, PVS.; P.S., Shijin Kumar. In: 2021 International Conference on Computer Communication and Informatics (ICCCI) Computer Communication and Informatics (ICCCI), 2021 International Conference on. :1-4 Jan, 2021;

[3]. Reyes, Andres F.; Camacho, Edgar C.; Armando, Mateus; Calderon, Juan M.. In: 2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC) Consumer Communications & Networking Conference (CCNC), 2021 IEEE 18th Annual. :1-6 Jan, 2021;

[4]. Sarang Shaikh; Sher Muhammad Daudpota; Ali Shariq Imran; Zenun Kastrati. In: Applied Sciences, Vol 11, Iss 869, p 869 (2021); MDPI AG, 2021.

[5]. Liao, Weizhi; Wang, Yu; Yin, Yanchao; Zhang, Xiaobing; Ma, Pan. In Neurocomputing. 21 March 2020 382:188-195 Language: English. DOI: 10.1016/j.neucom.2019.11.074, Database: ScienceDirect