**EVALUATION OF AUTO-SUGGESTION MODEL FOR HINDI**

**Introduction:**

The aim of this project is to evaluate or compute the auto-suggestion model for Hindi language. Auto-suggestion means suggesting or predicting something without explicitly giving an input. In this project, we work on prediction in text message applications. In text message applications, when we input a word, the application suggests the next word to the user. Now, based on requirement, the user may or may not use the suggested word. If the user meets the requirements, then the model predicting the word is said to be a good model otherwise we use other models to predict the word.

Here comes the evaluation. To confer the model status as good or if need to be improved, we score it for every correct suggestion of a word. Considering the score, we confer the status. Model is also evaluated based on training data set.

**Week1 & Week2:**

Firstly, we chose a project that displays the meaning when we hover the pointer on a word in a PDF file. We had started the project to an extent. But, we are unaware that there is a similar application called “Word Web”, which relates our idea. After realizing it, we terminated it. Then as per your suggestion we started the current project, “Evaluation of Auto-suggestion model For Hindi”.

* It was a little complicated to understand the project. After understanding the project, as per your guidelines, we made a poster presentation of the project.
* Then, we have started research on Deep learning concepts like LSTM, RNN.

**Week3 :**

* I started gathering Wikipedia data. I researched on methods to pre-process and clean the Wikipedia data. I found beautiful soup package. Beautiful Soup is a Python package for parsing HTML and XML documents. But, it did not work accurately.
* After beautiful soup, I tried pre-processing using minidom package. As the data collected by me was about 38GB, it took a time of about 8 minutes to compile the code.
* Then, we used Element Tree library for pre-processing the data. Here, we used less dataset size, when compared to former dataset. It worked here.
* After reviewing the article suggested by you written by Priya Dwivedi, implemented pre-processing using Perl program. But, this also did not work.
* Finally, implemented pre-processing and cleaning of data with ElementTree library.
* Again with respect to the article of Priya Dwivedi, converted words to vectors to feed the model.

**Week4:**

* As the next step after conversion of vector to words seems difficult, we chose LSTM method and trained the method with the data.
* We found a difficulty while parsing Hindi language data to the algorithm. I experimented to convert Hindi language to English language and then feeding the translated English language to the algorithm. But, it became complicated.
* Finally, we fed the Hindi language data directly to the algorithm and obtained auto-suggestion words for the given testing data.