- clarifAI Report SkillUp 2.0 -

- **What is clarifAI:**

  The Real-time assisted writing system provides the ability to suggest predictions for the next word. This makes typing faster, more intelligent and reduces effort. It is a Natural Language Processing concerned with predicting the text given the preceding text. It can be used as a web application.

- **Methodology/Description:**

  Assistant system provides the ability to autocomplete words and suggests predictions for the next word. This makes typing faster, more intelligent and reduces effort. The implementation involves using a large corpus. The initial task will be to design a keyboard interface as a web app. The keyboard layout consists of all keys which are present on a physical keyboard. The keyboard's interface will show the top three predictions for a given word sequence and suggest word-completion. This interface will be achieved by designing in HTML, CSS and JavaScript.

  If the space key is not pressed, then the server returns the auto-completions/spelling correction possible for the given word by using which of the word predictions previously made starts with what the user is typing. If less than three such matches are found, the minimum edit distance module takes over and returns the remaining predictions, to make a total of three predictions at all times. These predictions are converted to JSON and sent to the keyboard module. The keyboard unwraps the JSON and puts the predictions over the keys.

  Models Considered for word prediction:

- **N-grams Model:** Probabilistic models are used for computing the probability of an entire sentence or for giving a probabilistic prediction of what the next word will be in a sequence. This model involves looking at the conditional probability of a word given the previous words.If we consider each word occurring in its correct location as an independent event. We might represent this probability as:

$$P(w_1, w_2 \ldots, w_{n-1}, w_n)$$

We can use the chain rule of probability to decompose this probability:

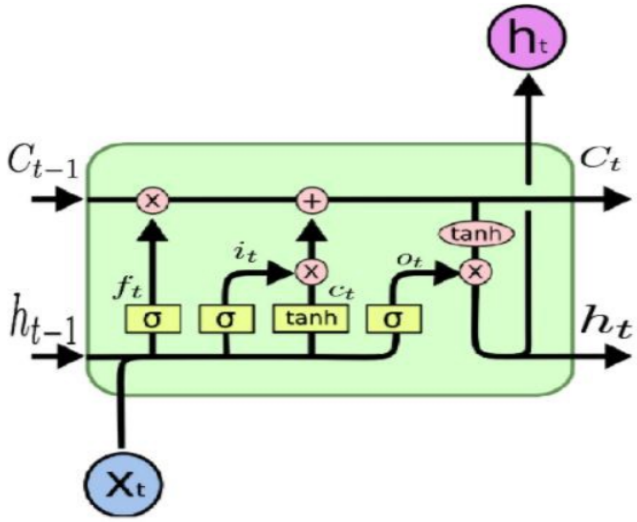
$$\begin{aligned} P(w_1^n) &= P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_n|w_1^{n-1}) \\ &= \prod_{k=1}^{n} P(w_k|w_1^{k-1}) \end{aligned}$$

Minimum Edit Distance:

$$P(t|c) = \min \begin{cases} distance[i-1, j] + ins\text{-}cost(target_j) \\ distance[i-1, j-1] + subst\text{-}cost(source_j, target_i) \\ distance[i, j-1] + ins\text{-}cost(source_j)) \end{cases}$$

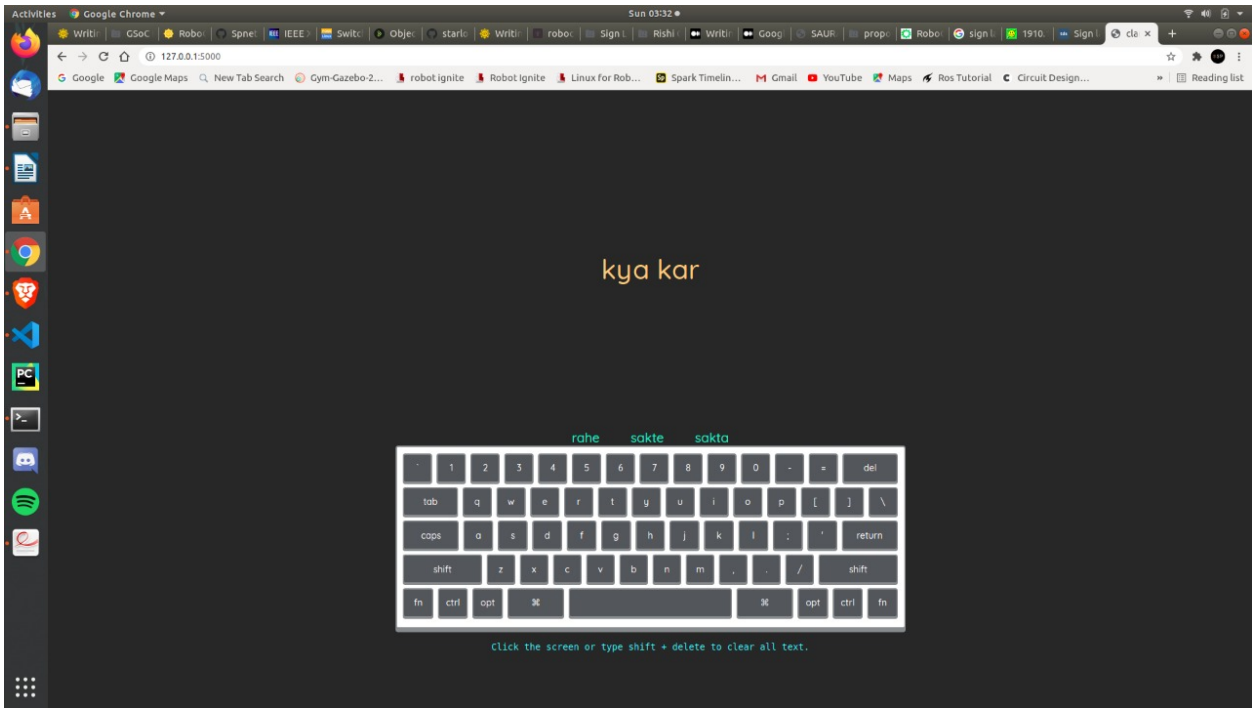
- **LSTM Model:** Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems. LSTM (Long Short-Term Memory) are very good for analyzing sequences of values and predicting the next one. The LSTM model uses Deep learning with a network of artificial “cells” that manage memory, making them better suited for text prediction than traditional

neural networks and other models. LSTM has the ability to remove or add information to cell states regulated by structures called gates.

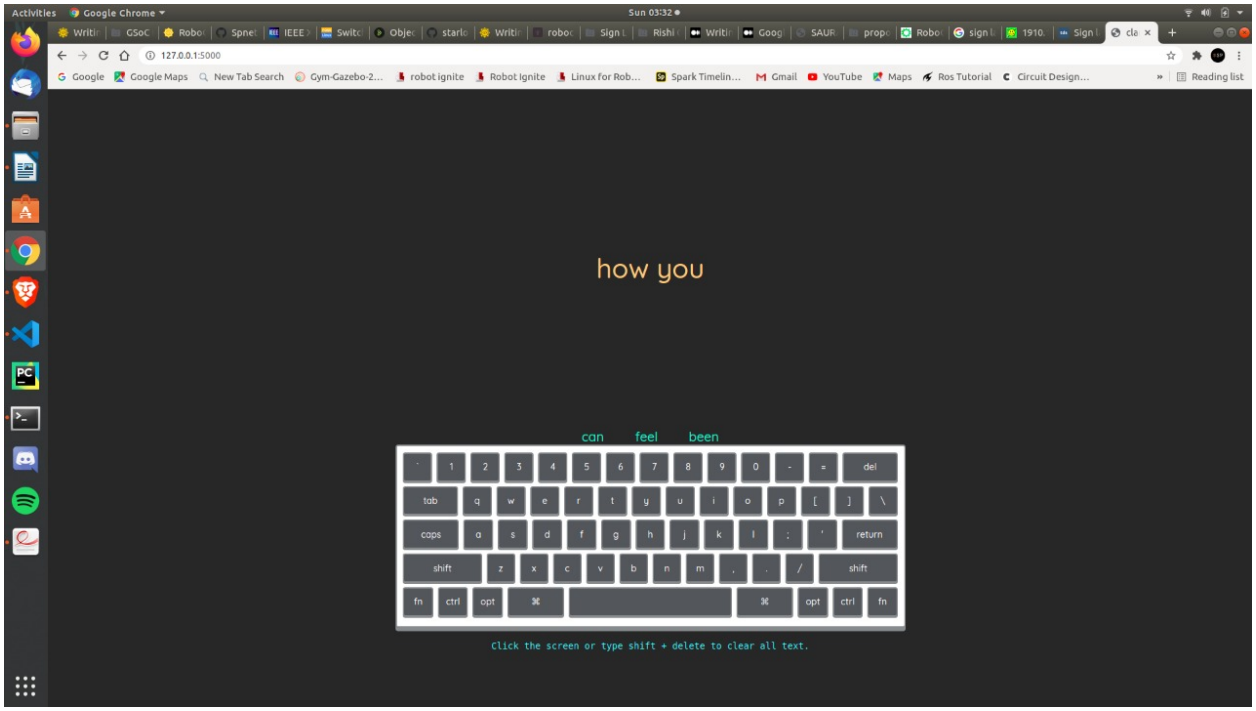


- **Technology Stack:**
  - Front-end - HTML, CSS, Js
  - Back-end - Flask, Python
  - Models - N-grams, LSTM
  - Frameworks - Keras, Tensorflow
  - Libraries - nltk, numpy

- **Working Example:**
  - Hindi:



  - English:



- **Datasets Used:**
  - https://www.kaggle.com/thanatoz/hinglish-blogs?select=data_119.txt
  - https://www.gutenberg.org/cache/epub/5200/pg5200.txt

- **Applications:**

  This project predicts the next possible word based on the input provided by the user. This project supports English as well as Hindi language.

- **Future Scope:**
  1. We could add an autocorrect feature like Grammarly.
  2. Try to improve our LSTM model to maximize the results for the word /phrase prediction.
  3. We could also implement an app where one can write an application letter and the app will predict the next appropriate words according to the subject of the letter.

- **Team Members:**
  1. Saurabh Suresh Powar
  2. Sakshi Chikshe
  3. Khushi Barjatia
  4. Rutuja Kolte

- **Mentor:**
  - Shreyansh Chheda

- **Acknowledgements:**
  - Deep Learning Specialization Andrew NG
  - Recurrent Neural Networks | Sequence Models | Natural Language Processing
  - COC VJTI