***Winter Internship Project Report***

***on***

***Roman to Devanagari Conversion***

|  |
| --- |
| cid:image001.jpg@01D1FC90.7D09CEE0 |
| **LeadingIndia.ai** |

|  |  |
| --- | --- |
| Submitted To: | Submitted By: Team Number – 20 |
| Dr. Kanad Kishore Biswas, **Professor** | Shiva Ganesh, Ramco Institute of Tech. |
| Shreyans Jain, **M Tech. Scholar** | Ambuje Gupta, Bennett University |
|  | Uday Agarwal, Bennett University |
|  | Monisha G., Bennett University |

**Department of Computer Science Engineering**

**School of Engineering and Applied Sciences**

**Bennett University**

**Greater Noida**

**Uttar Pradesh**

**Table Of Contents**

**Section 1 : Introduction ……………………………………………………….. 3**

* 1. **Introduction ………………………………………………………. 3**
  2. **Problem Statement ………………………………………………. 3**
  3. **Why we choose this ? ………………………………………………. 3**

**Section 2 : How to move Forward ………………………………………………. 4**

**2.1 Flowchart Diagram ……………………………………………… 4**

**2.2 Explanation with example ………………………………………. 4**

**Section 3 : Similar Projects ……………………………………………………... 6**

**3.1 What they have Done? ……………………………………………… 6**

**Section 4 : Proof Of Work ……………………………………………………… 6**

**4.1 Implementation ……………………………………………………… 6**

**4.1.1 Identifying Words ……………………………………… 6**

**4.1.2 Translating The Words ……………………………… 7**

**4.2 Screenshots ……………………………………………………… 8**

**4.2.1 Training Our Model ……………………………………… 8**

**4.2.2 Predicted answer for whether the word is English or Hindi .. 8**

**4.2.3 Translating the Sentences ………………………………… 9**

**Section 5 : Conclusions ………………………………………………………. 9**

**5.1 Leanings and Reflections ……………………………………… 9**

**5.2 Conclusions ……………………………………………………… 9**

**5.3 Limitations ……………………………………………………… 10**

**Section 6 : References ……………………………………………………… 11**

**Section 1 : Introduction**

**1.1 Introduction**

Whenever we talk on Facebook, WhatsApp, or Twitter we have seen that people are not very particular about expressing themselves in pure Hindi and pure English, but they tend to mix them up which troubles to a lot of people. So, we here are making a platform which will convert their mixed English and Hindi language to pure Devanagari form.Which will help them to make a better conversation and help them to increase their bonding. To make this happen we have made a dataset of English and Hinglish words each defined weather its english or hindi in front of them and their devanagari conversion of each.

**1.2 Problem Statement**

We have seen that instead of using pure English and pure Hindi languages while having a conversation on Social media or Tweeting a Tweet on Twitter people tend to mix up the languages which makes a lot of trouble for others. To handle these type of situation we here are making a platform which will convert these mixed up languages to pure devanagari form. Our tasks will be divided in 3 Objectives First, make a dataset of both english and Hinglish words and define them whether they are pure english words or hindi written in english and the other dataset which will contain the both english words and english sentences and their Devanagari conversion in front of them and the same for Hinglish words and Hinglish sentences. Second, let’s assume we have given a test sentence. Now, we will take each words from the test sentence and specify whether that word is English word or Hinglish word. So, Here we are tokenizing the sentence and specifying weather that word is English or Hinglish. Third, translating each tokenized word to Devanagari form and joining these words which will represent the Devanagari form the desired test sentence.

**1.3 Why we choose to do this ?**

Most people in India write Hindi in English language while tweeting on Twitter or chatting on Facebook and many people are not able to understand this. To help those we have build a platform which will convert whole text to one specific language (Devanagari) which help them to understand the conversation more clearly.

**Section 2 : How to move Forward**

**2.1 Flowchart Diagram**

Gather Dataset

Tokenize the words from test sentences

Identify whether the words is English or Hinglish

Convert Each Tokenized word to pure Devanagari form

**Fig 2.1(i) [Flowchart Diagram]**

**2.2 Explanation with an example**

Let our test sentence be I am going to work aur mei late ho jaaungi.

So, this a sentence which contain both pure english and hinglish words.

Step -1 : Gather dataset

Step -2 : Tokenize the words.

|  |
| --- |
| **Words** |
| I |
| am |
| going |
| to |
| work |
| aur |
| mei |
| late |
| ho |
| jaaungi |

Step -3: Identifying the words

|  |  |
| --- | --- |
| **Words** | **Identification** |
| I | en |
| am | en |
| going | en |
| to | en |
| work | en |
| aur | hi |
| mei | hi |
| late | hi |
| ho | hi |
| jaaungi | hi |

Step -4: Translate each word to their devanagari form and join them

Our output will be: मैं काम करने जा रहा हूँ और मुझे देर हो जाए

**Section 3 : Similar Projects**

**3.1 What they have done ?**

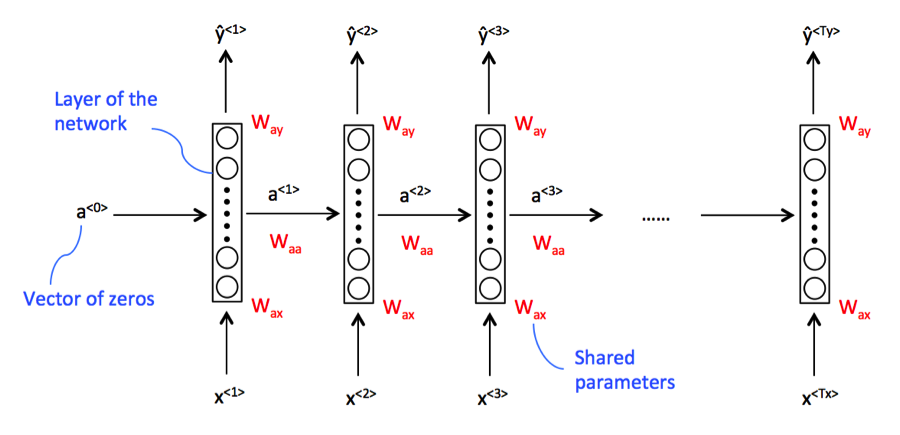
Many universities like IIIT Delhi[7], EFL University[8], NIT Agartala[9], IIIT Andhra Pradesh[9] has contributed in roman to Devanagari conversion. Our own dataset is also from IIIT Delhi we have requested them to use their dataset here [10].You can also read their artices how they have approached to these type of problems.What their article represents is Multi-Input Multi-Channel Transfer Learning based model (MIMCT) to detect offensive (hate speech or abusive) Hinglish tweets from the proposed Hinglish Offensive Tweet (HOT) dataset using transfer learning coupled with multiple feature inputs. Specifically, it takes multiple primary word embedding along with secondary extracted features as inputs to train a multi-channel CNN-LSTM architecture that has been pre-trained on English tweets through transfer learning. The proposed MIMCT model outperforms the baseline supervised classification models, transfer learning based CNN and LSTM models to establish itself as the state of the art in the unexplored domain of Hinglish offensive text classification

**Section 4 : Proof Of Work**

**4.1 Implementation**

**4.1.1 Identifying words**

To identify whether the word is english or hinglish we have our dataset of around 114235 words each specified whether the word is english or hindi using that dataset we trained our sequence model to identify weather the word is english or hinglish



**RNN (Recurrent Neural Network ) Model [6]**

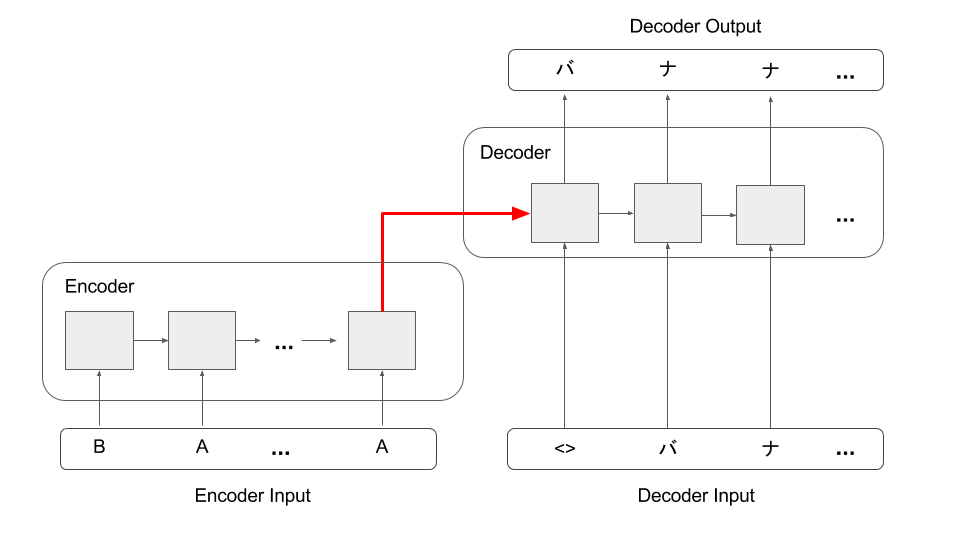
**Fig 4.1(i)**

The RNN scans through the data from left to right and the parameters used for each time step (Wax) are shared. The horizontal connections are governed by Wax parameters, which are the same for every time step. The Way’s are the parameters that govern the output predictions.

One very significant characteristic that we notice is that when making a prediction y, the network uses information not only from the corresponding input x, but also all from the previous ones. For example, for prediction ŷ<3> it gets information not only from x<3>, but also from x<1> and x<2>.

**4.1.2 Translating The Words**

To transliterate we used google tensorflow and seq2seq model. In that we taken the dataset and form that dataset we have stored the unique character in an array and using that unique character we have we vectorized the words which help us to translate the word effectively.



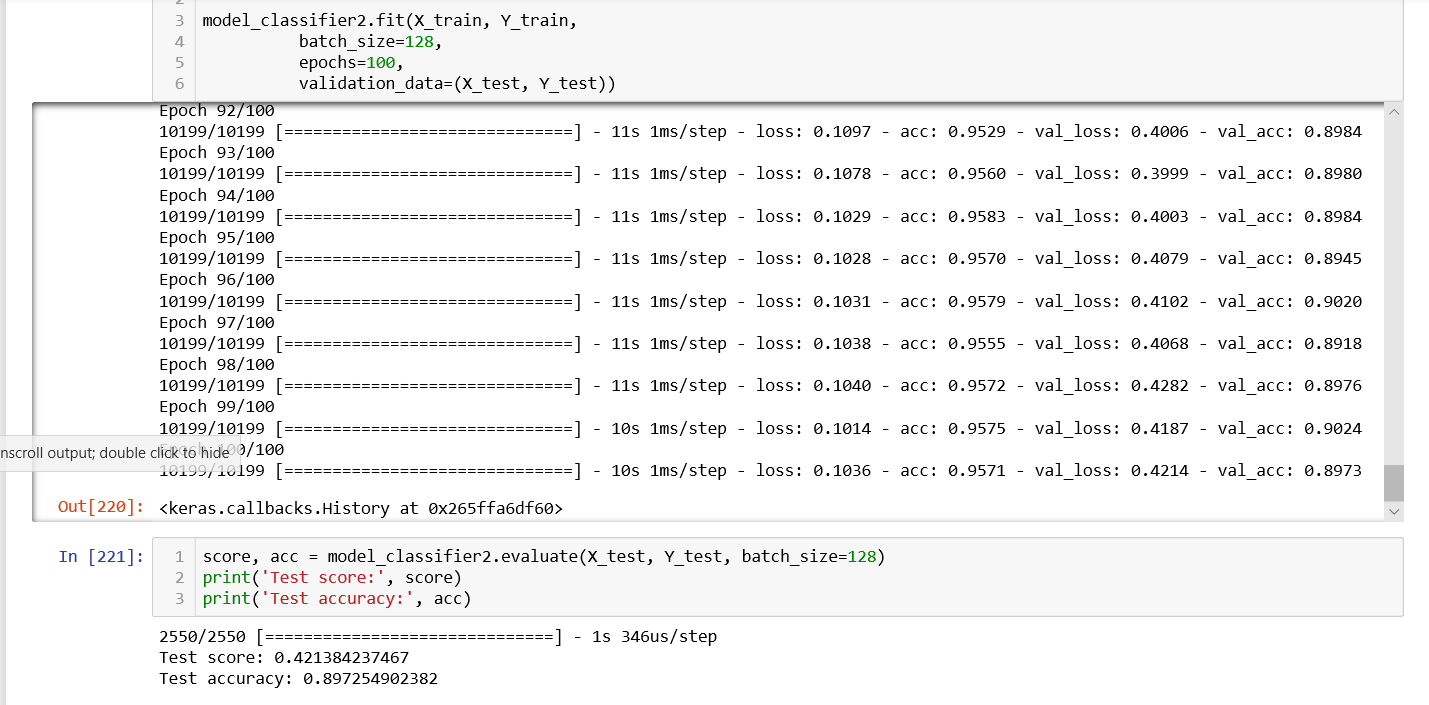
**Encoder-Decoder network [11]**

**Fig 4.1(ii)**

In the above figure, we can see the basic structure of a encoder-decoder model. The encoder encodes the words in a sequence and when the last word is read, it passes its internal hidden state to the decoder, which then starts generating the output sequence. Much of it is similar to a language model, however in language model, we take the probability distribution to choose the next word(sampling from the distribution), here we will use the greedy search and will select the next word using the highest probability in the softmax layer.

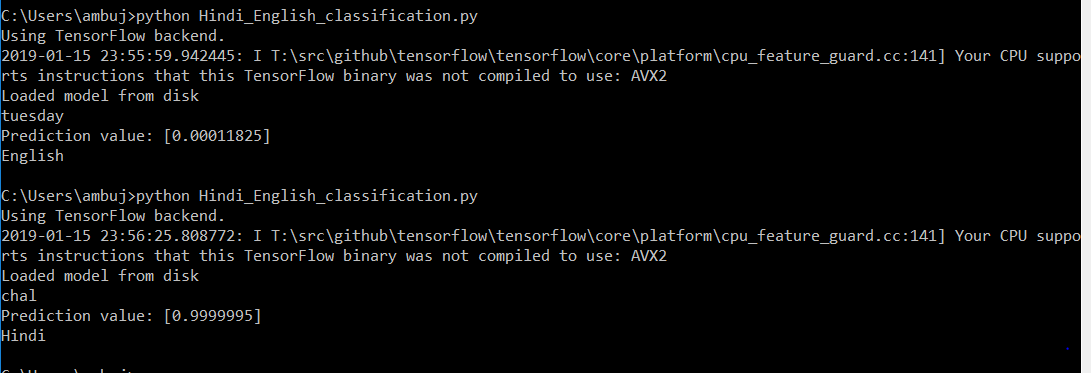
**4.2 Screenshots**

**4.2.1 Training our model**



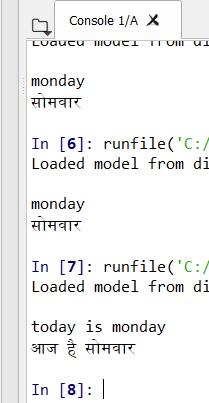
**Fig 4.2(i) [Model Training]**

**4.2.2 Predicted answer for whether the word is english or hindi**



**Fig 4.2(ii) [Predicting whether the word is English or Hindi]**

**4.2.3 Translating the sentences and words to Devanagari form**



**Fig 4.2(iii) [Translating the Sentences to Devanagari form]**

**Section 5 : Conclusions**

**5.1 Learnings and Reflections**

By this project we have learn how to work with keras, how to use different type of models, tensorflow which help us in their own domains. Many time it happens that these model did not work for us but keen determination and hard work help us to achieve our goal. We used keras to turn the tokenized words into the vector and then applying sequence models to identify each word whether it is english or hinglish and using seq2seq model to turn each tokenized word into their Devanagari form.

**5.2 Conclusion**

There are many different approaches and many different ways by which you can solve this problem and there are many other model available to do complete these steps. We chose this way or model because this was the most efficient way to do this problem.

**5.3 Limitations**

How our mind works? We humans always learn something new by interaction,or by watching videos but when a machine do our work like here a machine is converting codemixed sentenced to pure Devanagari which leads to some of the limitations.

1. We have provided it a limited amount of dataset so, it’s knowledge will always be limited to some boundary.
2. We cannot always guarantee the output shown by the machine will be always correct besides it will give some accuracy about the answer.
3. Our Project is limited to only one language whereas there can be many other language possible.

To overcome these limitations one can increase the size of the dataset and increasing the labels in it. Hence, we believe that as we move forward in future we will overcome most of the limitations and give people the more easy solutions to their problems.

**YouTube Link :** https://youtu.be/\_z6jFa5cbE0

**Section 6 : References**

**6.1 References**

[1] [https://github.com/danielv775/Natural-Language-Identification-Graduate-Project/blob/master/Language%20Identification%20System%20Graduate%20Project.ipynb](https://github.com/danielv775/Natural-Language-Identification-Graduate-Project/blob/master/Language%2520Identification%2520System%2520Graduate%2520Project.ipynb) (Havard Student)

[2] <http://machinelearningexp.com/deep-learning-language-identification-using-keras-tensorflow/>

[3] <https://towardsdatascience.com/tensorflow-gpu-installation-made-easy-use-conda-instead-of-pip-52e5249374bc>

[4] <https://stackoverflow.com/questions/51306862/how-to-use-tensorflow-gpu>

[5] <http://www.scitepress.org/Papers/2018/66786/66786.pdf> (Language-Identification)

[6] [https://www.google.com/imgres?imgurl=https://cdn-images-1.medium.com/max/1600/1\*WP0sqq0uyq\_bCGZJQTAb2g.png&imgrefurl=https://medium.com/machine-learning-bites/deeplearning-series-sequence-models-7855babeb586&h=435&w=900&tbnid=0dVqBOmPDpBdzM:&q=sequence+model&tbnh=103&tbnw=214&usg=AI4\_-kQmhR6al83iPe\_JVeVZ4v06bvz7MA&vet=12ahUKEwiqiqKm1PHfAhXBdH0KHdaPCA4Q9QEwAHoECAgQBg..i&docid=SzimUfrGNkAg7M&sa=X&ved=2ahUKEwiqiqKm1PHfAhXBdH0KHdaPCA4Q9QEwAHoECAgQBg](https://www.google.com/imgres?imgurl=https://cdn-images-1.medium.com/max/1600/1*WP0sqq0uyq_bCGZJQTAb2g.png&imgrefurl=https://medium.com/machine-learning-bites/deeplearning-series-sequence-models-7855babeb586&h=435&w=900&tbnid=0dVqBOmPDpBdzM:&q=sequence+model&tbnh=103&tbnw=214&usg=AI4_-kQmhR6al83iPe_JVeVZ4v06bvz7MA&vet=12ahUKEwiqiqKm1PHfAhXBdH0KHdaPCA4Q9QEwAHoECAgQBg..i&docid=SzimUfrGNkAg7M&sa=X&ved=2ahUKEwiqiqKm1PHfAhXBdH0KHdaPCA4Q9QEwAHoECAgQBg)

[7] <http://aclweb.org/anthology/W18-5118>

[8] <https://www.researchgate.net/publication/283090889_Real-time_Sentiment_Analysis_of_Hindi_Tweets>

[9] <http://www.scielo.org.mx/pdf/cys/v20n3/1405-5546-cys-20-03-00425.pdf>

[10] <http://precog.iiitd.edu.in/news.html>

[11] https://www.google.com/search?client=firefox-b-ab&biw=1920&bih=966&tbm=isch&sa=1&ei=Euo-XNTWK9q3rQGZ9puoCA&q=seq2seq+encoder+to+decoder+model+&oq=seq2seq+encoder+to+decoder+model+&gs\_l=img.3...45203.45203..45436...0.0..0.123.123.0j1......1....1..gws-wiz-img.1bFGTk51dOQ#imgrc=7AbEjxQb9YTIFM: