**INTERNSHIP REPORT**

**ICFOSS, Thiruvanathpuram**

**Name**: Govind G S

**College & course**: B.tech, Sree Chitra Thirunal College of Engineering

**Area** : Natural language Processing

**Duration**: 18-06-2018 to 13-06-2018

**Projects**: 1. Onscreen keyboard Malayalam

1. Malayalam transliteration using CRF

**Experience at ICFOSS**

The internship experience in ICFOSS gave me a better insight to idea of community development and importance of open source in promotion of technology.

**Workshop attended**

* FOSS Wednesdays is an idea worth reciprocating in colleges to share knowledge
* LoRa, the radio communication used in IOT
* Drone Technologies , Drone mapping to visualize a geographical location.

**Knowledge gained**

* Introduction to Gnome development and also to use GTK libraries.
* Wider knowledge in machine learning using CRF++ and an introduction to NLP concepts .

**Project reports**

1. **Onscreen keyboard in Malayalam**

Duration: 20/062018 to 3/06/2018

**AIM** : Development of an Onscreen Keyboard in Malayalam using GTK libraries in C.

The project was aimed at development of a Gnome onscreen keyboard in Malayalam also with the aim of typing in Malayalam for the disabled(who cannot use a keyboard).The project is developed using the gtk3 version 3.2 (https://developer.gnome.org/gtk3/3.22/)..

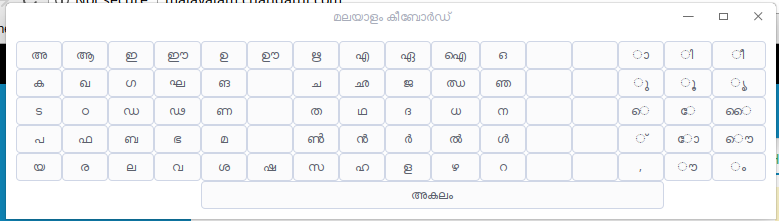
**Output**

A keyboard layout is developed for Malayalam and the keyboard will return the Unicode of the characters in the terminal screen .

The Github link : https://github.com/govind4873/malayalam-onscreen-keyboard-.

* **Scope of Improvement**
* 1. finding a method to input the Malayalam text to the input stream and not from the terminal
* 2. code optimization of the keys, instead of individual initialization of objects
* 3. function to return the Unicode from the button label instead of triggering individual functions .
* 4. Making it an standalone gnome application .

**Screen Shot**



2.**Malayalam Transliteration**

Duration: 4-07-2018 to 13-07-2018

**Aim:** To find the comparative efficiency of Malayalam to English and English to Malayalam transliteration using CRF

* We have used CRF++ for implementation of the CRF algorithm ,it is a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting/labeling sequential data. CRF++ is designed for generic purpose and will be applied to a variety of NLP tasks.
* The initial training set contains only 450 character sets. For the comparison the same training data is used with changed input and output and feature extractions.

**English to Malayalam Transliteration**

The task is to transliterate English to Malayalam, the existing methods use rule based systems that map each letters to corresponding English letter(s) and look up words that already exist in the database and provide collections

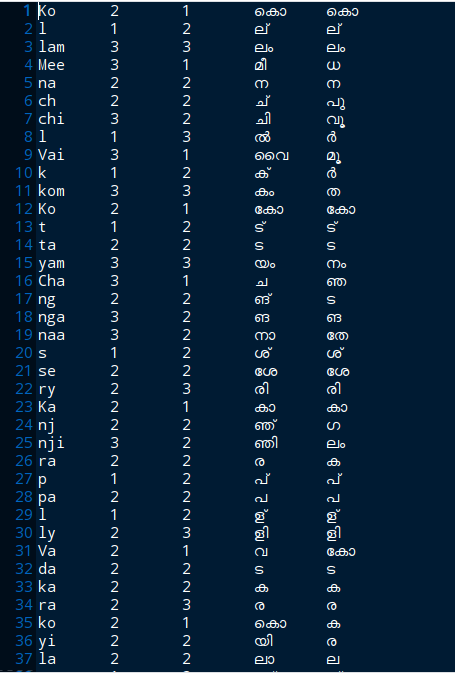
Challenges faced :

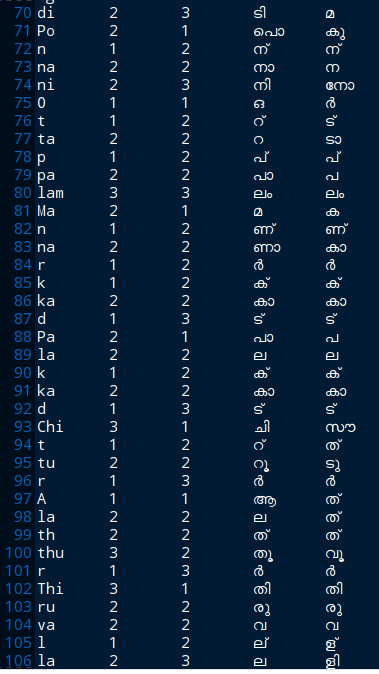
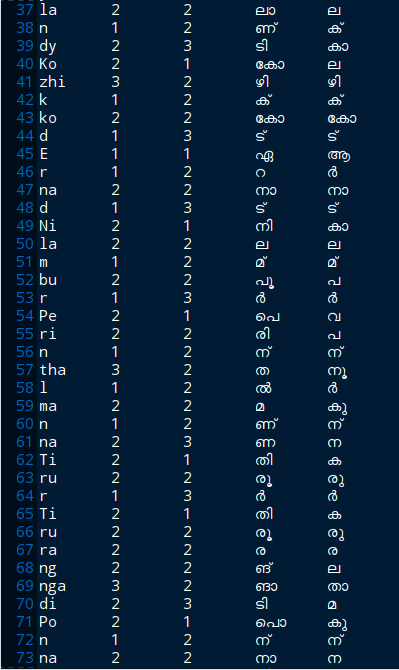
1. To find the corresponding features that allows CRF to map input and output. We have come up with the features
2. The number of letters in English to denote each pronounceable sets of words in Malayalam.
3. The position of the pronounceable set in the word spit to train the CRF
4. As we are training the data set with manually splitted words the model developed also need input such manner which is to be splitted by a program while the transliterated data is inserted and we were unable to develop an generalized algorithm to split English word to corresponding Malayalam pronounceable sets .

We could train and predict the letter sets with a test data of 100 sets and gets an accuracy of 40%

**Screen Shot**

The First row is the input Second row the letter count Third is the position of the Malayalam letter in the word Fourth is the output required and Fifth row is the predicted output.





**Malayalam to English Transliteration**

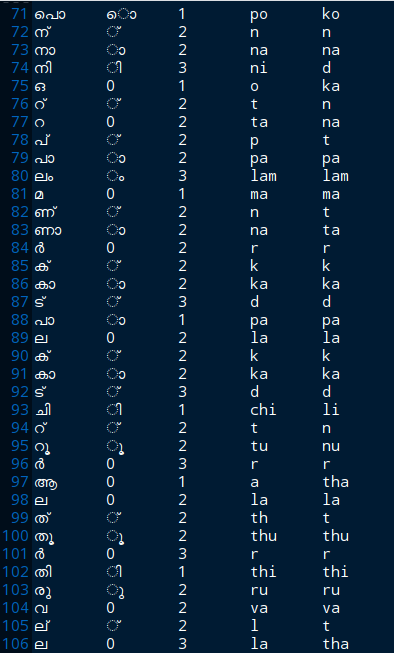
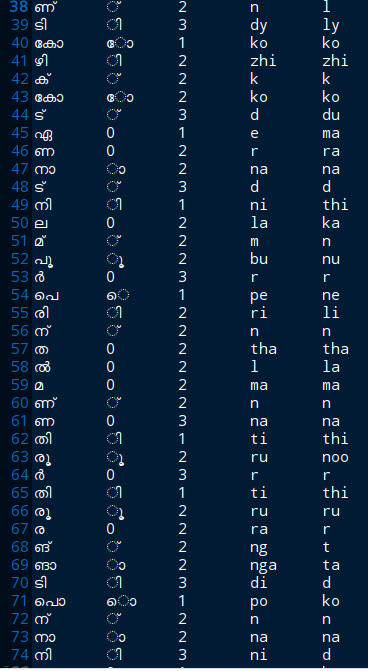
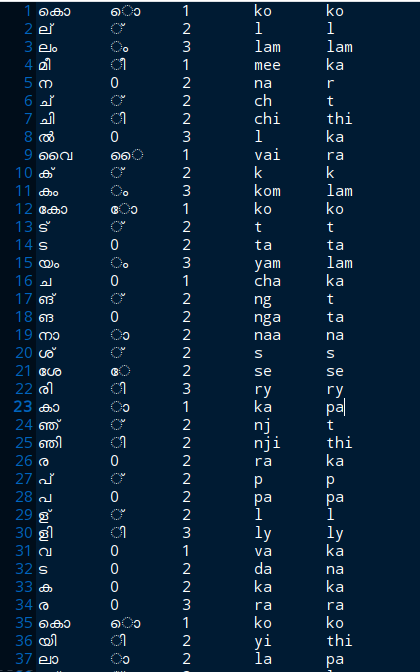
Malayalam to English transliteration is comparatively easier than the previous one as we can split the data of the input file with a generalized rule for splitting and for training.

Features used for training

1. The symbol accompanying the main letter (dheerkam).
2. The position of the letter in the word.

We could train and predict the letter sets with a test data of 100 sets and gets an accuracy of 65% and interestingly even the wrong predictions are comparatively similar in pronunciations than when observered with English to Malayalam .

Screen Shot



Output

* Malayalam to English transliteration is not recommended with CRF as its hard to find features that can be useful predict the output.

So it is better to train the data in to neural network to get better output and results.

* Whereas in case of Malayalam to English transliteration the possibility is higher to get better output in CRF as the relatable features are more strong and can be found. With a better training dataset it is possible to get better output than neural networks.