NLP Assignment 1 Report

Ujjwal Kishor Sahoo (21293)

Here we have been given two types of problems; one is Code mixed text classification, and the other is a Sequence labeling task.

Code Mixed Text Classification

Here we have been given three types of the dataset, namely hate, humour and sarcasm, and there are two tags where 0 corresponds to No hate/humour/sarcasm and 1 means hate/humour/sarcasm. I started by reading the dataset and understanding what the dataset is made up of so that I could apply all sorts of techniques to further improve the model. In the hate dataset, I used random over sampler and character level n-grams and applied them in different models. The best model was Random Forest with under-sampling, which gave the F1 score of 55% on tag 1. The best-performing model in the humour dataset was Random Forest with Random-Oversampling, which gave the F1 score of 74% in tag 1. In the sarcasm dataset, the best-performing model was Random Forest with Random-OverSampling, which gave the F1 score of 82% in tag 1.

Sequence Labeling Task

Using the Viterbi algorithm and a variant for noisy data, I examined how well the Hidden Markov Model (HMM) performed in sequence tagging. I calculated the word emission counts for every word or tag, the tag transitions, and the initial tag probabilities during training. I then turned these counts into probabilities. Next, I used dynamic programming to predict the most likely tag sequence for a sentence, applying the standard Viterbi algorithm. In order to address noise, I included an extension that dynamically adjusts emission probabilities for unknown words based on their usage context and changes the top-k probability paths based on sentence length.

I used baseline Viterbi for clean data, Viterbi with noise handling for noisy data, and Viterbi without any noise handling for noisy data to test the model. Given that the training and test data were well matched, the baseline Viterbi algorithm performed well, achieving 88.68% accuracy on clean data. Additionally, I am receiving an accuracy of 80.52% for the noisy dataset. In conclusion, while the noise-handling algorithm's extension has shown promise in processing unknown or noisy elements, it still has drawbacks when it comes to handling extremely noisy data.