PROJECT PROGRESS REPORT

Sandeep Nanjegowda – sgn3 (**Captain**)

Sunitha Vijayanarayan - sunitha3

Valentina Mondal – vmondal2

1. Progress made so far

Models Tried:

- Logistic Regression
- Naïve Bayes
- LSTM (Long-short term memory)
- GRU (Gated recurring units)
- CNN (Convolutional Neural Network)
- SVM (Support Vector Machine)

Pre-processing & Feature extraction:

- Have tried word embeddings like glove and word2vector on the models: LSTM, GRU and CNN.
- New features like word counts, emojis, hash tags were extracted and using for Logistic Regression and Naïve bayes.
- Both test and training set data was initially processed to remove html tags, punctuations, numbers, single characters, multiple spaces.
- Models were trained and fitted using the given twitter trained set data by splitting it into training and test set. Further the model was applied on the test data set to find out the precision, recall and F-Score.

More Details about models tried:

- **Logistic Regression:** Logistic Regression model performed well on the test set data and gave precision of 0.658, recall of 0.808 and F-score of 0.725. Were able to beat the baseline using the LR model.
- **Naïve Bayes**: Naïve Bayes did well on test data and gave F-Score of 0.702
- **Neural Network Models – LSTM, GRU & CNN**: We tried 3 neural network models LSTM, GRU, CNN with different combinations of hyper parameters. The maximum F-score obtained in all these neural network methods with different combinations of hyper parameters was 0.69. We tried word embedding with glove and word2vec but got almost the same results and not able to beat the baseline. Also tried without Glove and word2vec by training on all the words, but that model performed well only in local tests and did poorly in live data lab.
- **SVM:** For SVM, after using cross validation and grid search to find the good hyperparameter for SVM model, still the model could not beat the baseline with the test set data. Got an F-score of around 0.68.

2. Remaining tasks

- We have extracted additional features such as word counts, number of has tags, number of emojis in each of the tweet and used them in Logistic Regression. We are planning to use these features in Neural Network Models.
- Project documentation and presentation.
- We are looking at adding additional pre-processing steps like converting emojis and emoticons to text.

3. Any challenges/issues being faced

- Unable to beat the baseline (i.e., F-score of 0.723) in live lab leaderboard using LSTM, GRU and CNN.
- For neural network models like LSTM and GRU, overfitting is a problem as we get very good results on the graded test set in some cases but are not able to replicate the results on the leaderboard.
- Some models took very long time to train based on the hyper-parameters chosen.