NLP Final Project Proposal

Team Members: Jeffrey Hu, Ajinkya Patil

Project Dataset: [Natural Language Processing with Disaster Tweets | Kaggle](https://www.kaggle.com/competitions/nlp-getting-started/data)

Proposal Questions:

After you have selected a topic, a model, and a data set, submit a proposal of what you plan to do for the project. The proposal should be a few hundred words, and should address the following items. • What problem did you select and why did you select it? • What database/dataset will you use? • What NLP methods will you pick from the concept list? Will it be a classical model or will you have to customize it? • What packages are you planning to use? Why? • What NLP tasks will you work on? • How will you judge the performance of the model? What metrics will you use? • Provide a rough schedule for completing the project.

For our project, we selected a text classification problem in an ongoing Kaggle competition for fake disaster tweets. We’ve selected this problem, due to the fact that currently, twitter/X is having issues authenticating their users and their resulting tweets, with the release of twitter blue/X premium. Now that fake ‘parody’ accounts of real people/organizations can now get ‘blue check marks’ despite their inauthenticity, verification of truth on twitter is harder than ever. Thus, the verification of real/authentic tweets is more relevant than ever, and this dataset mixes fake disaster tweets and real tweets, something that could have serious consequences if a reader is misled. We will use the kaggle competition dataset, given in the introduction. We will utilize Pretrained NLP transformer models to help predict the label, and will decide to compare the results with Rule Based classical Models and Recurrent Networks. We plan on using nltk/spacy/any of the other packages to help with EDA and possible tokenization/text cleaning/basic sentiment analysis/feature creation for classical models, and PyTorch for transfer learning/getting the pretrined NLP transformer models, or to create our own RNN model architecture. Our NLP task is text classification. The performance of the model can be judged by categorical metrics like accuracy, ROC/AUC, and f1 score.

Our plan/schedule, up until the presentation on December 11th, goes like so:

1. Nov 6: Project Proposal and Literature Review
2. Nov 13: Data Loading/Cleaning/Preprocessing, EDA
3. Nov 20: Model procuring (transfer learning) + training
4. Nov 27: Model evaluation/comparison (Pretrained vs. Classical vs. RNN)
5. Dec 4: Finishing up/Presentation Preparation
6. Dec 11: Presentation