Project
Project Progress Report as of 11/16
CS410 Text Information Systems
Fall 2023

Project Team

- Tahir Bagasrawala (tahirib2)
- Ashwin Saxena (ashwins2)
- Aryan Gandhi (aryang6)
- Abrielle Agron (aa106) (Captain)
- Harish Venkata (hkv2)

What tasks have been completed?

- ✓ Finalized project objective to "*Leaderboard Competition Creation using Natural Language Processing with disaster tweets dataset.*"
- ✓ Completed design specification and project plan tasks to execute below:
    - Configure Leaderboard using *LiveDataLab*.
    - Connect project GitHub to Leaderboard in *LiveDataLab*.
    - Establish approach to compare models using a common evaluation criterion accurately and fairly.
    - Identify a list of NLP models to evaluate.
    - Start with OkapiBM25 – *review results, pros, and cons*.
    - Evaluate K-means clustering – *review results, pros, and cons*.
    - Evaluate LDA – *review results, pros, and cons*.
    - Evaluate LSA / PLSA – *review results, pros, and cons*.
    - Evaluate out-of-the-box APIs (e.g., *TextBlob*) – *review results, pros, and cons*.
    - Evaluate SciPy's pre-built NLP packages – *review results, pros, and cons*.
    - Provide Conclusion and Final Recommendation
    - Document Final Project Report
- ✓ Configured Github repository – added documentation for project proposal.
- ✓ Identified Kaggle's Natural Language Processing with Disaster Tweets to predict which tweets are about real disasters and which are not.
- ✓ Configured initial Leaderboard in *LiveDataLab*.
- ✗ Attempted to connect our project GitHub repo to Leaderboard in *LiveDataLab* by using previously developed code in MP 2.2.
    - TA's *Mu-Chun Wang* and *Yuxiang Liu* provided guidance to not pursue this integration to *LiveDataLab* – due to complexities in implementation.
    - The project team have incorporated their feedback to establish a baseline score and train and validate several NLP models against the baseline to beat the baseline.

    - Requested *Mu-Chun Wang* to delete the initial Leaderboard created.
- ✓ Developed approach to compare classifier models using a common evaluation criterion accurately and fairly in python. Configured the following helper functions:
    - *load_and_preprocess_data (csv_path)*
    - *load_model (model_path, class_name)*
    - *evaluate_model (model, X_test, y_test)*
    - *Leverage sklearn metrics for accuracy, precision, recall, and F1 score*
- ✓ Deployed a baseline and 6 additional classifier models from *sklearn* toolkit.
    - Logistic Regression Model *(baseline)*
    - Random Forest Classifier Model
    - AdaBoost Classifier Model
    - Decision Tree Classifier Model
    - K-Neighbors Classifier Model
    - Gaussian Naïve Bayes Model
    - Gradient Boosting Classifier Model
- ✓ Document Project Progress Report (this document).

What tasks are pending?

- Evaluate Support Vector Classification Models – *review results, pros, and cons*.
- Evaluate LDA – *review results, pros, and cons*.
- Evaluate LSA / PLSA – *review results, pros, and cons*.
- Evaluate out-of-the-box APIs (e.g., *TextBlob*) – *review results, pros, and cons*.
- Evaluate SciPy's pre-built NLP packages – *review results, pros, and cons*.
- Perform Hyperparameter Optimization (HPO) for all approaches.
- Provide Conclusion and Final Recommendation
- Document Final Project Report

Are you facing any challenges?

- The team was struggling to connect the project GitHub repo to a Leaderboard in LiveDataLab – were looking for example code and documentation.
    - No longer an issue as advised by *Mu-Chun Wang* and *Yuxiang Liu.*
- The team is now unblocked to continue evaluating NLP models against our dataset.