Tweet-Analyzer

This project results in the sentiments of people regarding to any event happening in the world by analyzing tweets related to that event. It will search for tweets (here datasets are provided) and analyze each tweet to see how positive or negative it's emotion is.

A separate <training> and <testing> dataset is provided.

Built using:

Python 3.6

Dataset Information:

We use and compare various different methods for sentiment analysis on tweets (a binary classification problem). The training dataset is expected to be a csv file of type tweet_id,sentiment,tweet where the tweet_id is a unique integer identifying the tweet, sentiment is either 1 (positive) or 0 (negative), and tweet is the tweet enclosed in "". Similarly, the test dataset is a csv file of type tweet_id,tweet. Please note that csv headers are not expected and should be removed from the training and test datasets.

Requirements:

There are some general library requirements for the project and some which are specific to individual methods. The general requirements are as follows.

numpy
scikit-learn
scipy
nltk

Preprocessing:

- Run stats.py <preprocessed-csv-path> where <preprocessed-csv-path>. This gives general statistical information about the dataset and will two pickle files which are the frequency distribution of unigrams and bigrams in the training dataset. (this step is already done and the files exist in the directory).

After the above steps, you should have four files in total: <preprocessed-train-csv>, <preprocessed-test-csv>, <freqdist>, and <freqdist-bi> which are preprocessed train dataset, preprocessed test dataset, frequency distribution of unigrams and frequency distribution of bigrams respectively.

- Run python3 tAnalyser.py for predictions of your model to be saved in decisionTreeResult.csv.

Note: Set TRAIN = True while training and then change it to False while testing.

Tweet-Analyzer

Files:

- tAnalyser.py (Main file with preprocessing and decisionTree algorithm)
- util.py (Additional Funtinalities used within the algorithm)
- stats.py (General statistical information about the dataset and will two pickle files which are the frequency distribution of unigrams and bigrams in the training dataset.)
- decisionTreeResult.csv (The final results are stored here)
- ./datasets/ (Directory containing all the required data for the model)

Built By: iimashfaaq & prachidhingra09