**Data:**

Raw data:

davidson.csv

wassem\_css.csv

wassem\_hovy\_naacl.csv

Merged\_Raw\_Data.csv

combining non-preprocessed ‘Text’ and ‘isHate’ columns from 3 raw data datasets

proprocessed\_full.csv

Preprocessed Merged\_Raw\_Data.csv

Train\_1\_gram\_prep.csv

1-gram encoded training data (with 6000 feature columns and ‘Label’ column)

Validate\_1\_gram\_prep.csv

1-gram encoded holdout(validate) data (with 6000 feature columns and ‘Label’ column)

Train\_2\_gram\_prep.csv

2-gram encoded training data (with 6000 feature columns and ‘Label’ column)

Validate\_2\_gram\_prep.csv

2-gram encoded holdout(validate) data (with 6000 feature columns and ‘Label’ column)

Train\_3\_gram\_prep.csv

3-gram encoded training data (with 6000 feature columns and ‘Label’ column)

Validate\_3\_gram\_prep.csv

3-gram encoded holdout(validate) data (with 6000 feature columns and ‘Label’ column)

**Code:**

N-gram feature extraction.ipynb

Construct N-gram feature model on train and holdout(validate) set using N = 1,2,3

Ngram + SVM\_increase\_RAM.ipynb

On N-gram feature models, train linear SVM and non-linear SVM models with different regularization hyperparameters.

Ngram + Gradient boost.ipynb

On N-gram feature models, train Xgboost Classifier with parameters tuned on n\_estimators and min\_sample\_split

BERT.ipnb

Using pre-trained bi-directional encoder and a NN infrastructure to train the classifier

Character-level encoding + LSTM.ipynb

Using character-level Tokenizer and pad\_sequences to convert each text sentence into a 586 dimensional array, and then pass it to LSTM model for training