JIGSAW TOXIC COMMENT CLASSIFICATION

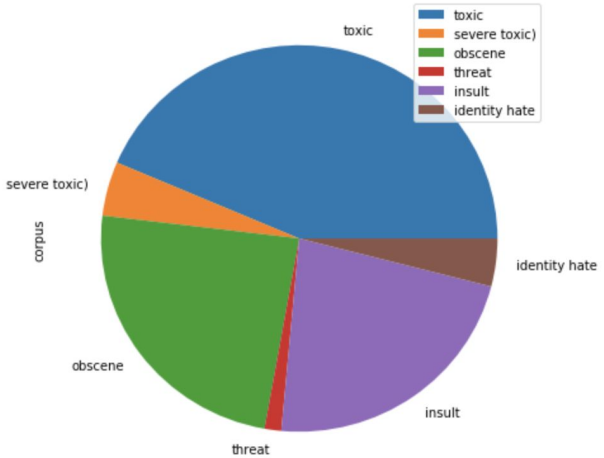
Group member: Amy, Nilan, Claire

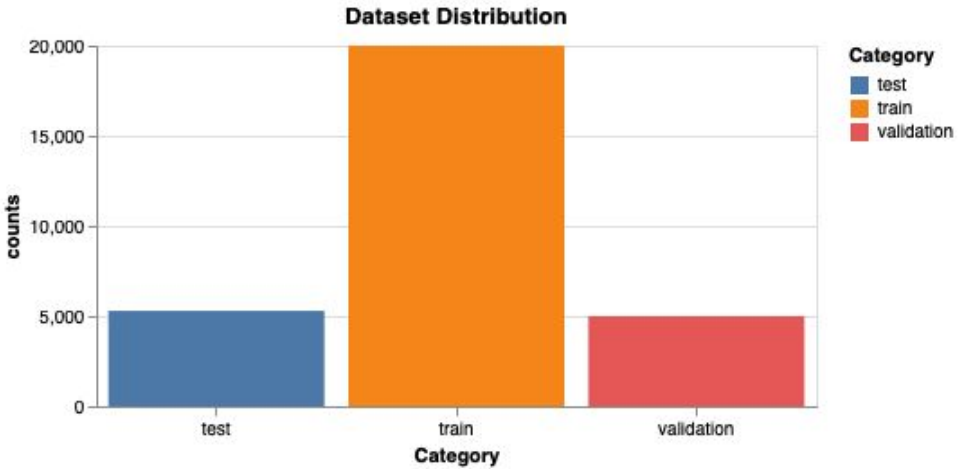
MOTIVATION

- People stop sharing different opinions because of the threat and harassment online.
- More and more platforms is seeking facilitate conversations for their users.
- Discussing things in a diverse but friendly environment.

Goal: Detect toxic comments and classify different types of toxicity.

DATA


toxic
severe toxic)
obscene
threat
insult
identity hate
toxic
severe toxic)
corpus
identity hate
obscene
insult
threat


Dataset Distribution
20,000
15,000
10,000
5,000
0
counts
test
train
validation
Category
Category
test
train
validation

PREVIOUS WORK

- Detecting and Classifying Toxic Comments *by Kevin Khieu and Neha Narwal*
  - Applied Support Vector Machine(SVM), Long Short-Term Memory Networks(LSTM), Convolutional Neural Networks(CNN) and Multilayer Perceptron(MLP)
- Challenges for Toxic Comment Classification: An In-Depth Error Analysis *by Betty van Aken et a*l
  - Combined Logistic Regression, bidirectional RNN, bidirectional GRU with Attention layer and CNN, with pretrained word embeddings from Glove and sub-word embeddings from FastText
- Automatic Toxic Comment Detection Using DNN *by D'Sa et al*
  - Apply BERT onto feature-based CNN and RNN models with an regression-based approach
- Offensive Language Identification and Categorization with Perspect and BERT *by Pavlopoulos et al*
  - BERT performed better in categorizing the offensive type

METHODS

- Not a typical binary or multi-class problem
- 6 binary classifiers for each individual class
- Computation Overload
- BCELoss (Binary Cross Entropy)

Toxic	Severe_toxic	Obscene	Threat	Insult	Identity_hate
0	0	0	0	0	0
1	1	1	0	1	0
1	0	0	1	0	0
1	1	0	0	1	1

METHODS

- **Baseline** - Embeddings + Unidirectional LSTM Network + Linear Layer
- **Main Network**
    - BERT Embeddings
    - Pooler Output i.e. last layer hidden-state of the first token (CLS)
    - Dropout
    - Linear Layer
- 1 if prediction > 0.5 else 0

EXPERIMENTS & RESULTS

- Experiments
  - Batch size: 16, 32
  - Epochs: 2, 3, 4
  - Learning rate (Adam): 5e-5, 3e-5, 2e-5
  - Max_grad_norm: 0.7, 0.8, 0.9, 1.0, 1.1
  - Warmup_proportion: 0.05, 0.1, 0.15, 0.2
- Results
  - Baseline LSTM: Macro-F1 Score of 0.86199 (epoch:5; batch size:32; learning rate :0.001)
  - BERT: 0.90505 (best combination is highlighted)
- Result on Kaggle test set
  - The graded score: 0.97905
  - Ranking on the public leaderboard: 2586/4500+ entries

CONCLUSION AND FUTURE WORK

- Conclusion:
  - Best combination of our model is batch size of 32, epochs of 2 and learning rate of 2e-5 which have F1-score of 0.905.
  - BERT model performs aproximately 4% better than the baseline LSTM model
- Future Work
  - Study more on dealing with data imbalance
  - Extend to apply XLNet pre-trained model