Toxicity Classification On Social Media Platforms

Aakarshan Dhakal adhakal@stanford.edu
Dhruv Kedia dkedia@stanford.edu

Project Overview

Motivation

With the increasing use of social media platforms, there has been a sharp increase in toxic comments on online platforms. It is important that toxicity in online platforms is reduced so that the internet is a safe place for people no matter their gender, age, race, nationality, likes, dislikes, religious beliefs, or political beliefs.

Problem Statement

The input to our algorithm are comments from an online platform. We then use a deep convolutional neural network architecture that uses character to sentence information to output a predicted toxicity class (for example: identity hate).

Applications

Filtering Posts and Comments
Social Media Policing
User Education

Data

- Uses the Toxic Comment Classification Challenge dataset from a Kaggle competition.
- Dataset included 159,571 Wikipedia comments that have been hand labelled by human raters for toxicity levels.
- Different classes of toxicity: toxic, severe toxic, obscene, threat, insult, and identity hate.
- Over 140,000 comments are non-toxic
- Toxicity Class Distribution is depicted in figure 1
- A comment can belong to multiple toxicity classes
- Correlation between different toxicity classes depicted in figure 2
- Data Split - Train:Validation:Test, 80:10:10
- Dataset Example:
  - Comment - “Last Warning! Stop undoing my edits or die!”
  - Toxicity Class - Toxic, Threat

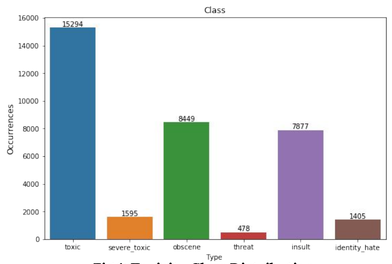

Fig 1: Toxicity Class Distribution

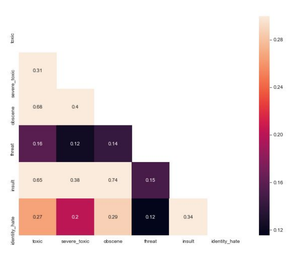

Fig 2: Correlation heatmap of genres

Approach

- Our model is based on the model described in the paper Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts by Cicero Nogueira dos Santos.
- Model:
  - Takes a sentence as an input and computes a score for each of the sentiment labels.
  - To compute the scores, the model passes the sentence through a sequence of layers that extract features with increasing complexity levels.
  - The model extracts features at the character level, word level, and sentence level.
  - The model architecture uses two convolutional layers.
  - A visual representation of the model can be seen in figure 3 and the convolutional approach can be seen in figure 4.
- Binary Baseline Model
  - Easy to analyze and train
  - Loss Function - softmax
- Multi-label Model
  - Single model is ideal and allows for correlations to be learned
  - Loss Function - linear logistical loss functions for multiple classes

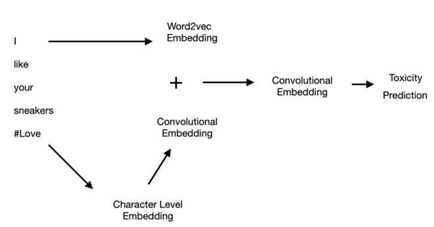

Fig 3: Model overview

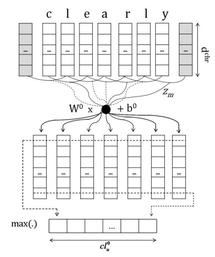

Fig 4: Convolutional Approach

Experiments And Results

Evaluation Metrics

- Accuracy, Precision, Recall - Binary case
- Average Precision (AP) - Multi-label case
- AUC Score (Area under the ROC Curve)

Word Embeddings

- GloVe word embedding vectors (trained on twitter data) VS Training word Embeddings from scratch.
- Results:
  - Higher Average Precision for GloVe
  - Lower Training Time for GloVe

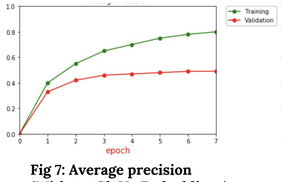

Fig 7: Average precision (Without GloVe Embeddings)

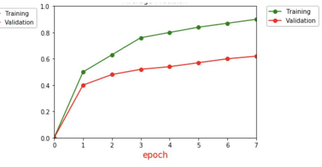

Fig 8: Average precision (With GloVe Embeddings)

Dropout

- Model without drop overfits as seen in fig 5
- Use of dropout helps in regularization (fig 6)
- Best dropout probability = 0.4

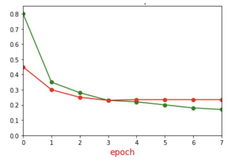

Fig 5: Loss without dropout

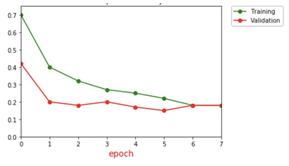

Fig 6: Loss with dropout probability 0.4

Best Model Results

Our best model uses pretrained GloVe word embeddings, dropout of 0.4, learning rate of 0.1 and an Adam optimizer.

Dataset	Avg. Precision	AUC score
Validation	0.62	0.9782
Test	0.65	0.9740

Qualitative Analysis

- Direct insults and comments are tagged toxic with respective labels with a very high confidence. For example, whenever, a sentence contains words such as "f*ck, shit, shut up, idiot".
- Toxic comments that had spelling errors were also labelled as toxic with a fairly high confidence. Examples: ". F* ck ing trollreasons" "if ytou think shes greek your a morooon."
- Certain words were strong indicators of severe toxicity. For example - "d*ck", and "assh*le". This was problematic because a sentence about "Moby Dick" was classified as severely toxic when in reality it had no element of toxicity.
- The model fails to identify a statement as non toxic when the language used is strong. For example: Have you tried to fathom why reactions to you are harsh?" The presence of strong words such as fathom, and harsh confused the model into incorrectly classifying the sentence as toxic.
- Statements that contain insults in foreign languages or unfamiliar slangs are not labelled accurately. For example: “chamars” in Hindi means “untouchables”, and the model could not pick on this as identity hate.

Conclusion

- Best multilabel model uses a customized loss function, GloVe word embeddings, and employs dropout.
- Based on AP, multilabel model performs averagely.
- Our AUC of 0.9740 is not much lower compared to the best model on the Kaggle Leaderboard that obtains an AUC score of 0.98856.
- Key limitations working on this project was training time. It prevented us from further hyper-tuning several of the parameters.

Future Work

- Better hypertuning of parameters.
- Integrating BERT and ELMO contextual word embeddings with our model.
- Experimenting with a LSTM + GRU based architecture.
- More in depth error analysis.

References

1. Toxic Comment Classification Challenge, Kaggle, www.kaggle.com/c/jigsaw-toxic-commentclassification-challenge.
2. Dos Santos, Cicero & Gatti de Bayser, Maira. (2014). Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts.
3. Gong, Yunchao, Jia, Y, Leung, T., Toshex, A., Ioffe, S. Deep Convolutional Ranking for Multilabel Image Annotation. \textit{arXiv:1312.4894}.
4. Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation.