Toxic Comments Classification (Identity Bias)

Surbhi Prasad
Karishma Chauhan
Jun 30, 2022

AGENDA

1. Situation and Data Source

Overview of goal and data

2. EDA and Preprocessing

Distribution of words in toxic comments and core distt

3. Evaluation Metric

Metric getting minimized including bias component

3. Model Comparison

Models tried and compared to pick best

Situation

For non-toxic comments, model predicts as toxic with a higher rate. Models predicted a high likelihood of toxicity for comments containing those identities (e.g., “gay”), even when those comments were not actually toxic (such as “I am a gay woman”).

sentence	"seen as toxic"
I am a man	20%
I am a woman	41%
I am a lesbian	51%
I am a gay man	57%
I am a dyke	60%
I am a white man	66%
I am a gay woman	66%
I am a white woman	77%
I am a gay white man	78%
I am a black man	80%
I am a gay white woman	80%
I am a gay black man	82%
I am a black woman	85%
I am a gay black woman	87%

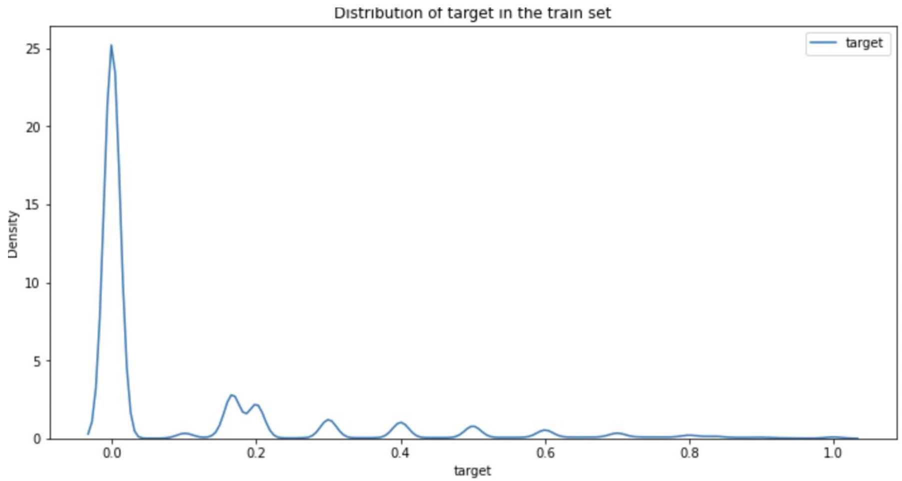
Build a model that recognizes toxicity and minimizes this type of unintended bias with respect to mentions of identities.

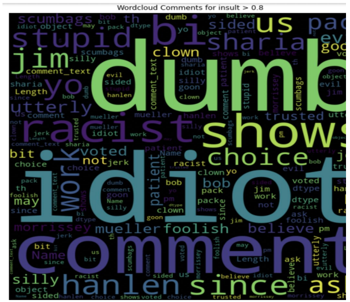
At the end of 2017, the Civil Comments platform shut down and chose to make their ~2m public comments from their platform available in a lasting open archive

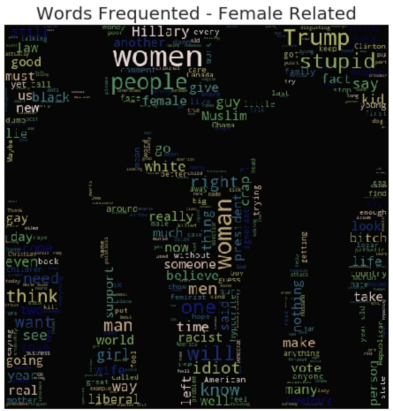
Relevant identities: ***male, female, homosexual_gay_or_lesbian, Christian, Jewish, Muslim, black, white, psychiatric_or_mental_illness.***

Exploratory Data Analysis

There are almost 70 % of data has target values<=0.1







92 % of data belong to the non-toxic class and 7 % of data belong to the toxic class

In all subgroups, there are 77.55 % of comments have NAN values.

Text Comments PreProcessing

Lower and upper latter, numbers, extra space, http/https links,

Punctuations, emojis, other languages like Chinese, meaningless words

which may not be found in English dictionary

- Remove those words (Non-English words).
- contraction of the word, removing of extra space, removing of Punctuations, http/https links, removing stop words excluding NOT word

- Using sequence bucketing, we can speed this up by dynamically padding every batch to the maximum sequence length which occurs in that batch

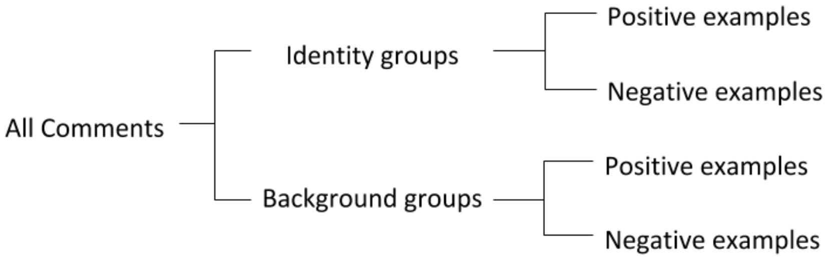
Evaluation Metric

$$\text{Score} = w_0 \, AUC_{overall} + \sum_{a=1}^{A} w_a M_p(m_{s,a})$$

where, A = number of submetrics (3)

$m_{s,a}$ = bias metric for identity subgroup s using submetric a

w_a = a weighting for the relative importance of each submetric (w values set to 0.25)



a. Subgroup AUC — This calculates AUC on only the examples from the subgroup. It represents model understanding and performance within the group itself

b. BNSP AUC — This calculates AUC on the positive examples from the background and the negative examples from the subgroup.

c. BPSN AUC — This calculates AUC on the negative examples from the background and the positive examples from the subgroup.

Power mean of all three bias metrices

Baseline Model : TFIDF + SGD

Binary Classification Problem where we have to classify a given comment as toxic or Non-toxic.

TFIDF with SGD Classifier for different alphas were tried,, Best alpha=0.0001
Penalty L2, log loss
TFID parameters:
Ngram range: (1,2), Min df: 3, smoothing

```
For values of alpha =  1e-05 The auc score on CV is: 0.810816
For values of alpha =  0.0001 The auc score on CV is: 0.8254805333333333
For values of alpha =  0.001 The auc score on CV is: 0.8034773333333332
For values of alpha =  0.01 The auc score on CV is: 0.7876821333333334
For values of alpha =  0.1 The auc score on CV is: 0.7841066666666666
For values of alpha =  1 The auc score on CV is: 0.7866538666666668
For values of alpha =  10 The auc score on CV is: 0.7896832
```

Deep Learning Model: LSTM + GRU

- Text Padding to 200 words
- KFold=3
- Transformed sentence to seq of words
- Glove embeddings for Vocab, max =500000
- Batch Size=512
- Epochs=2
- Embedding vocab size: 404791

- Log Loss Function
- Adam Optimizer

Hidden Layer size= 64, 32, layers=2, dropout=0.2

Layers

1. Bidirectional LSTM
2. Linear
3. GRU
4. GRU +LSTM
5. Output Linear

```
Epoch 1: Train loss: 0.4442, BIAS AUC: 0.9021, Valid loss: 0.4323, BIAS AUC: 0.9214
Epoch 2: Train loss: 0.4280, BIAS AUC: 0.9229, Valid loss: 0.4275, BIAS AUC: 0.9231
```

```
Epoch 1: Train loss: 0.4451, BIAS AUC: 0.9016, Valid loss: 0.4311, BIAS AUC: 0.9179
Epoch 2: Train loss: 0.4285, BIAS AUC: 0.9225, Valid loss: 0.4273, BIAS AUC: 0.9237
```

```
Epoch 1: Train loss: 0.4450, BIAS AUC: 0.9026, Valid loss: 0.4319, BIAS AUC: 0.9190
Epoch 2: Train loss: 0.4282, BIAS AUC: 0.9224, Valid loss: 0.4283, BIAS AUC: 0.9207
```

Future Scope and Challenges

1. Text Augmentation to include other resources
2. Ensemble of BERT and LSTM Models
3. Methods like improved Bucket Sequencing to fasten training
4. Improved weighted loss function as per subgroups
5. Improve AUC further by more preprocessing like converting emojis to words

etc.

THANKS