Detecting Toxicity in Online Forums

Amy Chen, Farzaan Kaiyom, Kristen Anderson
{amyjchen, farzaank, ander50n} @stanford.edu
CS230 Spring 2019

Motivation

Reduce toxicity on the web with fewer inequity-based errors

↓

Create a classifier using model-bias limiting techniques

Examples*:
"f*** women" → Toxic
"u suck at coding cuz ur a girl" → Toxic
"I am a woman" → Not toxic
"being a woman sucks" → Not toxic

*These examples are less extreme than what is in our dataset.

Data

kaggle

Civil Comments Data
Jigsaw Unintended Bias in Toxicity Classification Competition. *Team: CS230*

Labeled for toxicity and identity by annotators. Labelling scheme (by severity) allow for mild but not extreme toxicity. 29% of data labeled for identity.

Our Team

Identity-mentioning text from online

Encyclopedia articles, sentence generators, news articles and editorials found by us, from known non-toxic sources. Labeled using identity keywords.

Features

Comment Text
String → Vectors
Length: 1 to 1906 words
Tokenized + Vectorized
GloVe word embeddings

Processed for sequential models

Identity labels
Boolean
24 labels for race, religion, sexuality, gender, disability

Labeled examples given more weight in training

Toxicity Labels
Float → Boolean
Target label
>= 0.5 are toxic (true/1)
< 0.5 are not toxic

Predicted value (the goal)

Metadata/misc
Not used
Annotator count, time posted, likes, reacts, type of toxicity, etc.

We don't use these provided features.

Process + Models

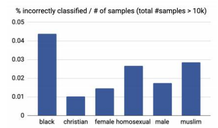
INITIAL MODELS

Baseline Model
To find a good baseline, we tested and tuned four models for accuracy.

Model	Test acc
2-layer NN	0.9089
LSTM	**0.9505**
CNN	0.9249
CNN-LSTM	**0.9507**

The LSTM and CNN-LSTM performed well (~0.95), but the **CNN-LSTM ran significantly faster.**

OPTIMIZATION FOR ACCURACY & BIAS + ERROR REDUCTION TECHNIQUES



Error Analysis

Wrote script to identify identity distributions and find **error-inducing keywords.**
Ex: insane, crazy, silly.

Samples with **black, muslim, and homosexual** identity mentions were **disproportionately misclassified.**

Data Augmentation

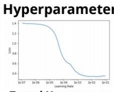
Accuracy
Included commonly toxic words in non-toxic context.

Bias Reduction
Used news/editorials/encyclopedia entries for non-toxic use of identities, annotated with a script detecting keywords. Based on our research [1]

Auxiliary Labels + AUC

Trained models to predict auxiliary labels (identity labels) to improve predictions by identity [2]. **Labels were based on AUC by subgroup, which we evaluated our models on** (see Results).

Hyperparameter Search



Tuned Hyperparameters Learning Rate and Epoch Number through creating learning rate vs. loss graphs for multiple epoch numbers

GloVe Word Vectors

Supplemented limited vocab from training set using GloVe pre trained word vectors

Used **Common Crawl** 840b token vectors, which most closely matched our expected usage

FINAL MODELS

LSTM
Lr: 0.001
Epochs : 4
Batch size: 512
maxlen=300

- Embedding layer with spatial dropout
- 2 stacked LSTM Layers
- 2 hidden linear layers (relu; input: LSTM layers, max and avg pool)
- 2 relu output (1 for targets, 1 for auxiliary labels)

LSTM/Attn
Lr: 0.001
Epochs : 4
Batch size: 512
maxlen = 220

- Embedding layer with spatial dropout
- 2 stacked LSTM Layers
- 2 attention layers
- 2 hidden linear layers (relu; input: attention layers, max and avg pool)
- 2 relu output (1 for targets, 1 for auxiliary labels)

BERT
Lr: 0.00002
Batch size: 32
Epochs : 1

Trained Google's pretrained and uncased BERT model with 69.4% of the training data to train and export a model to use on the test data.

Ensemble Averaging LSTM + BERT

Used a rank ensemble model with a weight of .65 on our LSTM and .35 on the BERT model

$$\hat{y} = \sum_{i=1}^{N} \omega_i y_i(\mathbf{x})$$

where $\hat{y}$ is the final prediction and ω is the probabilistic weight of each model's output y_i.

Results

Model DA: Data Augmentation	Train (acc)	Train DA (acc)	Train (auc)	Train DA (auc)	Test (auc)	Test DA (auc)
Baseline CNN-LSTM	0.9501	0.9643	0.9618	0.9645	**0.9067**	**0.9080**
LSTM	0.9632	0.9612	0.8769	0.8929	**0.9364**	**0.9359**
LSTM/ATTN	0.9541	0.9549	0.8188	0.8139	**0.9346**	**0.9339**
BERT					**0.9365**	
Rank Ensemble BERT + LSTM					**0.9391**	**0.9392**

BERT improved by 3% after marginal increases in dropout and increased exposure to the training set
Both LSTMs were a 3% improvement over baseline, although our LSTM with attention performed worse than expected.
Data Augmentation improved our baseline and rank ensemble model, but not our LSTMs.
Rank Ensemble worked better than Linear Ensemble by 1%

Train: 1624387 **Val:** 180996
Aug Data: 4580 **Test:** 97320

AUC SCORE EQUATIONS [3]

$$M_p(m_s) = \left(\frac{1}{N}\sum_{s=1}^{N} m_s^p\right)^{\frac{1}{p}}$$

M_p = the pth power-mean function
m_s = the bias metric m calulated for subgroup s
N = number of identity subgroups

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

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; all four w values set to 0.25

Discussion

The Good: **Auxiliary labels** helped significantly— it likely made our model distinguish between ways how people talk about identity (versus other subjects). **Embeddings** gave our models a broader vocabulary. **Rank Ensemble** gave us the best of both our models.

The Bad: Our **LSTM with Attention**. As a model we forked, we suspect that the tradeoff between sequence length and attention didn't pay off.

The Okay: **Data Augmentation** provided mixed results. Confusing examples of people talking about harassment or other negative experiences may have caused this. **Hyperparameter Tuning** was insightful but inactionable given competition time limits.

Future Work

Error analysis on our other models to figure out data analysis flaws. **Supplement Data** with more examples for lesser represented identities, even distributions by identity and comment length.

Further model improvements: A better LSTM/attention model, train BERT on augmented data, and ensemble average the resulting model

References

[1] Jeffrey Sorensen, Nithum Thain, Lucy Vasserman, Lucas Dixon, John LI. 2018. Measuring and mitigating unintended bias in text classification. *Jigsaw.*
[2] Quang Nguyen, Hamed Valizadegan, and Milos Hauskrecht. 2011. Learning classification with auxiliary probabilistic information. In *2011 IEEE 11th International Conference on Data Mining*, pages 477–486. IEEE.
[3] Jigsaw. 2019. Jigsaw unintended bias in toxicity classification.
Thank you to the entire Kaggle community for their tips and educational examples.