Abuse isn't always physical!

An approach to Toxic Comment Classification

Pratyush Singh
University of Zurich
pratyush.singh@uzh.ch

Rohit Kaushik
University of Zurich
rohit.kaushik@uzh.ch

***Abstract*—Abuse isn't just physical. The online abuse in terms of toxic comments on social media or Wikipedia's talk page are becoming more and more regular. With the developments in machine learning and natural language processing, it is possible to identify such hate comments and, thus also identify the abuser for criminal justice. In this paper, we present three learning models, namely a simple linear baseline based on logistic regression, a bidirectional gated recurrent unit based convolutional network with GloVe embeddings, and a BERT model. We report the standard F1 scores and AUC scores for the models and compare our scores to one of the best research done in this topic and report relative increase of 4% accuracy. We were able to achieve accuracy as high as 98% with the BERT model but with a trade-off. Our CNN-BiGRU model achieves accuracy of 97%. Overall, our models improve upon the previous results and we find the CNN-BiGRU model to be the best one.**

***Index Terms*—BERT, CNN, text classification, toxic comments**

I. Introduction

In today's world, social media and online discussions have become a common part of our life because of it's easiness to reach out for interactions and services and goods. But social media platforms are also prone to online abuse. Online abuse covers a wide range of behaviours and technologies. Abuse happens when someone acts in a way that causes harm and distress to others. It is often obvious that someone is behaving in an abusive way but it's not always clear where the boundary falls between expressing a point of view and being abusive. In extreme cases, cyberbullying can even lead to victims committing suicide. With the developments in language technology and machine learning, it has become increasingly important to tackle online abuse and identify the criminals. Though, identifying criminals is out of scope of this project but we provide fascinating results on the toxic comment classification part. In this project, we use the data available from Kaggle competition (here) to identify the toxic comment into one of these subgroups: toxic, severe_toxic, obscene, threat, insult and identity_hate. The dataset collected have been labelled by human raters for the toxic behavior. We train three models: a simple linear baseline based on logistic regression, a bidirectional gated recurrent unit based convolutional network, and a BERT model.

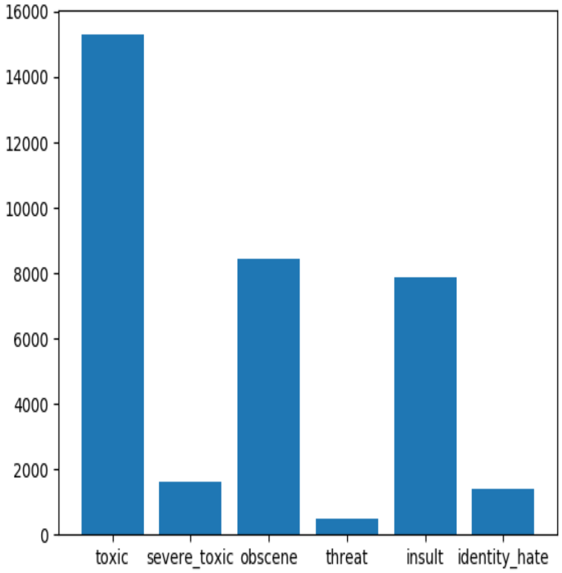
The paper is organised as follows: In section II, we present the related work done for the task of toxic comment classification and our motivation to choose the above neural architectures. The section III describes the dataset and the section IV presents our all three approaches with details about parameters. In section V, we provide the experimental results and reasoning for the obtained results and also compare our results to one of the best work done so far. The section VI provides information about the ablation study performed during the project and is followed by a conclusion section.

II. Related Work and motivation

One of the first work in comment abuse classification with machine learning was completed by Yin, et al.'s paper, in which the researchers use a support vector machine and apply TF-IDF to the features [5]. Our work trains a strong linear baseline using logistic regression inspired from [5]. Yahoo [4] implements an automated algorithm for filtering online toxic comments using a combination of parser, lexical and syntactic features to train a supervised classifier. In most of the recent works, Chu et al. [2], researchers have used Recurrent neural network (RNN) with Long short-term memory (LSTM) cells as a sequence model to achieve good results. Using RNNs alone has a disadvantage because of the vanishing gradient problem in long term dependencies as pointed out by Bengio et al. [1]. Researchers in [2] also train a Convolutional neural network (CNN) with word embeddings and character embeddings. The CNN model with character embeddings showed best results for them. Taking motivation from this, we decide to train a CNN based Bidirectional Gated recurrent units (GRUs) combining the benefits of both and the results of which are beyond expectations. Since not a lot of work has been done in this regard by using Bidirectional Encoder Representations from Transformers (BERT) [3] architecture and there is no reference material on this, we also used pre-trained BERT models and fine-tuned it for the task of toxic comment classification and the results, as expected, was better than all our results.

III. Dataset

The dataset is created by the Conversation AI team, a research initiative founded by Jigsaw and Google are working on tools to help improve online conversation. The dataset is from Wikipedia's talk page edits. In this dataset, there are 160K comments and labeled with different categories (toxic, severe_toxic, obscene, threat, insult and identity_hate) some of the comments belong to more than one category. Our model predicts a probability of each type of toxicity for each comment ID. We train on 159571 comments and test on 153164 comments. There were few empty comments which we got rid of in the data preparation phase by using Natural Language Toolkit (NLTK) to remove punctuations, stop words etc.



The above figure describes our training dataset. Our dataset was moderately imbalanced with sever_toxic & threat & identity_hate having fewer training examples. We tried to deal with the class imbalance by assigning class weights in the objective function.

IV. Proposed Approach

Our method consists of three models for comment classification:

- A Logistic Regression (Strong linear Baseline)
- A Bidirectional GRU with convolutional neural network
- Pre-trained BERT model

A. Strong linear Baseline

With this model, we create a strong baseline for this task using sklearn's logistic regression. As mentioned earlier, our method addresses the problem using a supervised learning approach. This model uses naive bayes and bag of words approach. We create a bag of words representation, as a term document matrix and use ngram range from 1 to 2 to tackle the problem at hand. The features are made of word ngrams. It turns out that using the term frequency–inverse document frequency (TF-IDF) gives better priors than the binarized features. The TFIDF weight for term *i* in post *j* is:

$$TFIDF_{ij} = TF_{ij}.IDF_i$$

Term frequency provides a measure of how important a particular term is in a given post (a local weighting). It is defined as:

$$TF_{ij} = \frac{n_{ij}}{\sum_k n_{kj}}$$

where n_{ij} is the number of occurrences of term *i* in post *j* and the denominator is the count of the occurrences of all the terms in post *j*. TF_{ij} will be larger for terms that appear more often in a comment. The *IDF* term provides a measure of how important a particular term is within the entire corpus (a global weighting). IDF for term *i* is defined as:

$$IDF_i = log\frac{|P|}{|p_j : t_i \in p_j|}$$

where $|P|$ is the total number of posts and $|p_j : t_i \in p_j|$ is the number of posts in which term t_i appears. Terms appearing in many posts will have lower IDF scores.Terms that appear in the corpus but not in a given comment will recieve a TFIDF weight of 0. We use the default L2 norm as the penalty measure with dual formulation for logistic regression. The regularization strength of 0.25 was used and other parameters can be looked up in the code.

B. Bidirectional GRU with convolutional neural network

Convolutional Neural Network : CNNs have provided astonishing results in the field of Natural Language Processing. For each word there is a row of vectors of fixed dimension. By using convolution, ngram can be generated just like a sliding window of different size passed all over the words. Using property of GloVe that similar words have similar cosine distances and cosine distances are similar to dot products and the dot product is actually a convolution. From the pair of the word embeddings, convolutional filters will learn meaningful features. For text, we use 1-D convolution because we can slide the window in only one direction. Padding is required, so that the size of the input and the output is the same. Apply max-pooling, taking the maximum activation value, output from the convolution passing through the whole text. Next we apply more dense layers and multi-layered perceptron on the top of these features and train it for classification task.

Bidirectional GRU Like humans, machines also predict well on the basis of past knowledge. It's an RNN but can tackle the vanishing gradient problem. Bidirectional GRU's are a type of bidirectional recurrent neural networks with only the input and forget gates. It allows for the use of information from both previous time steps and later time steps to make predictions about the current state. We use 128 units for BiGRU and the hidden layers are concatenated and passed through a feed forward neural network. A SpatialDropout layer is used after the Glove embedding layer. The output layer is a dense layer with 6 neurons and a sigmoid activation function. We use binary crossentropy as the loss function since it is best suited for multiclass classification problem. The optimizer used for training is Adam Optimizer.

GloVe : GloVe is an unsupervised learning algorithm for acquiring vector representations for words. GloVe model is trained on global collection of word-word co-occurrences statistics in a corpus, and the results show a linear substructure of words in vector space. The training objective of GloVe is to learn word vectors such that their dot product equals the logarithm of the words' probability of co-occurrence. Two different ways to apply Word Embeddings to a neural network

- Train embedding layer
- Use a pre-trained embedding (like,here, GloVe)

C. Pre-trained BERT model

Bidirectional Encoder Representations from Transformers (BERT) is a NLP pre-training technique developed by Google. It is based on the transformer architecture that makes use of attention for contextual representation of words in a sequence.

Since inception researchers have used BERT on several NLP problems and were able to achieve state of the art results on some including text classification. We were motivated by the strength of BERT and decided to use it for our task.

We make use of the framework Framework for Adapting Representation Models or FARM developed by deepset-ai. On top of the BERT embedding we use a single layer which is the prediction head and the loss function for this task is binary crossentropy. We train our Bert model with two different embedding size, 128 & 256.

V. Experimental Results

In this section we describe our experiment results, mainly the accuracy scores for different models 1) baseline 2) CNN-BiGRU 3) BERT

A. Baseline- Logistic Regression Classifier

As a baseline to compare our other models we decided to use Logistic regression classifier, since it is a simple yet powerful model for multi-class classification. We were able to achieve an AUC of 0.98 and an F1-score of 0.60.

B. Convolutional Neural Network with BiGRU

Our CNN-BiGRU model with GloVe embedding achieves an AUC of 0.99 and F1-score of 0.78. We find CNN to perform the best for this multi-class classification task. In the next section, we also show that smaller kernel size can learn better classification than larger kernel size.

Comparing our results to [2], our best model has F1-score of 0.78 and accuracy of 96-97% which outperforms the paper's best model of CNN with character embeddings which has F1-score of 0.57 and classsification accuracy of 94%.

C. BERT Classification

The below table describes class-wise accuracy metrics as well the average scores. Overall we were able to achieve F1-score of 0.69 and accuracy of 98% which is more than what achieved by researchers in [2]. This score is very close to our results from CNN-BiGRU and we believe while BERT is indeed quite powerful for this task it was a bit overkill and not necessary since simpler models achieves decent accuracy. Given the resource needed and time for training it was not feasible to train BERT changing parameters constantly and hence we also believe that BERT model could have been better tuned.

class	precision	recall	F1-score	support
toxic	0.56	0.91	0.69	913
severe_toxic	0.39	0.41	0.40	46
obscene	0.65	0.79	0.72	543
threat	0.75	0.41	0.53	37
insult	0.74	0.68	0.71	509
identity_hate	0.76	0.50	0.61	107
average	0.64	0.78	0.69	2155

TABLE I
BERT SCORES WITH EMBEDDING SIZE 128

class	precision	recall	F1-score	support
toxic	0.55	0.91	0.69	913
severe_toxic	0.39	0.30	0.34	46
obscene	0.64	0.79	0.70	543
threat	0.69	0.49	0.57	37
insult	0.70	0.69	0.70	509
identity_hate	0.76	0.50	0.61	107
average	0.62	0.79	0.68	2155

TABLE II
BERT SCORES WITH EMBEDDING SIZE 256

VI. Ablation Study

In this section we discuss qualitative evaluation of our models and how different parameters affect the model performance.

Our baseline logisitic classifier model takes tf-idf score of words as input for classification. To understand how this affect our model accuracy we use different ngrams, and find that increasing ngram range lead to poor results. Smaller n_grams suggests that local feature perform better for classification.

n-gram range	AUC	F1-score
1,2	0.98	0.60
1,3	0.977	0.60
1,4	0.97	0.58
1,5	0.96	0.54

TABLE III
BASELINE WITH DIFFERENT NGRAM RANGE

Similar inference can be made by using CNN kernel of different sizes. To understand the effect of CNN parameters on accuracy we perform ablation study on the kernel size. Smaller kernel size can learn more localized features whereas larger kernel size tend to learn more global features. And the localized features is needed for short comments to be classified accurately. As we see increasing the kernel size affect the model accuracy adversely.

Performing ablation study was difficult task on BERT model given the computational time and resource requirements. Training and testing the BERT model takes approximately

kernel-size	AUC	acc	F1-score
2	0.99	0.96	0.78
3	0.991	0.97	0.76
4	0.98	0.78	0.75
5	0.95	0.76	0.75

TABLE IV
CNN WITH DIFFERENT KERNEL SIZE

20 hours, hence it was not feasible to perform a thorough hyper parameter tuning. We train and test our BERT model on two different embedding sizes since we believe BERT embedding to be the most important parameter, in our control, for classification accuracy. We experimented with embedding size of 128 and 256 and didn't find any significant difference in the results.

embedding-size	AUC	acc	F1-score
128	0.995	0.98	0.69
256	0.993	0.97	0.68

TABLE V
BERT EMBEDDING SIZE

VII. Conclusion

Our best model with regards to multiclass classification was CNN-BiGRU using GloVe word embeddings as input. It outperforms the CNN model as described in the paper [2].

Our CNN model also performs better than BERT which we believe is because of poor hyperparameter tuning over BERT model which takes long time to train and test.

The Ablation study results suggest that smaller kernel which can learn more local features performs better for classification. This can be seen in table IV which suggests that the model performs worse with larger kernel size. Similar interpretation can be made from table V which shows increasing ngram range leads to poorer model scores.

In the future, we aim to perform a more in depth study of the transformer architectures like BERT and XLNet on classification tasks. We also intend to do a more thorough ablation study with more hyper parameter tuning.

Acknowledgment

We would like to thank Prof. Simon Clematide for giving us the opportunity to work on this project. We are grateful for his guidance and valuable inputs during presentations and otherwise discussions.

Code

The code to generate the results in paper can be found here

References

[1] Y. Bengio, P. Simard, and P. Frasconi. "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2 (1994), pp. 157–166.

[2] T Chu, Kylie Jue, and Max L. Wang. "Comment Abuse Classification with Deep Learning". In: 2017.

[3] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. arXiv: 1810.04805 [cs.CL].

[4] Chikashi Nobata et al. "Abusive Language Detection in Online User Content". In: *Proceedings of the 25th International Conference on World Wide Web*. WWW '16. Montréal, Québec, Canada: International World Wide Web Conferences Steering Committee, 2016, pp. 145–153. ISBN: 9781450341431. DOI: 10.1145/2872427.2883062. URL: https://doi.org/10.1145/2872427.2883062.

[5] Dawei Yin et al. "Detection of harassment on Web 2.0". In: (Jan. 2009).