

Jigsaw Rate Severity of Toxic Comments

Rank relative ratings of toxicity between comments

EE 541 Final Project- Fall 2021

Dheeraj Panneer Selvam, Rashaad Hussain

December 13, 2021

1.Introduction

Toxic behavior on social media is expected, but it becomes increasingly unacceptable. Toxicity within the social realm can be described as the spread of unnecessary negativity or hatred that adversely affects the people it encounters. Toxic people spread malicious intent on the Internet and try to abuse others in discussions. A study by (Kwak et al)[1]. showed toxic behavior in online team competition games. They found that the outcome of the match was related to the development of toxic behavior. Toxic comments on social sites such as Twitter can be found on topics that are extremely difficult to discuss, such as Brexit, climate change, abortion, vaccines, and the US presidential election. Toxic behavior is more common on such topics due to its divisive nature. When discussing such topics, people tend to have different opinions, which can lead to division (D. Yin et al., 2009)[2].

The ultimate objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable. Most NLP techniques rely on machine learning to derive meaning from human languages. Machine learning explores the construction and study of algorithms that can learn from and make predictions on data. Such algorithms operate by building a model for example inputs in order to make data-driven predictions or decisions, rather than following strictly static program instructions.

Since detection of Toxic comments and their classification has already in application in many online platforms, this project aims to rank the severity of the toxic comments. In this project, we will tackle the Jigsaw rate severity of toxic comments Challenge. When humans are asked to look at individual comments, without any context, to decide which ones are toxic and which ones are not, it is not an easy work. Also, each person will have a different threshold for toxicity. If Majority vote is taken to make a decision on it researchers have pointed out that this discards meaningful information. Therefore, it is a much easier to people decide which of two comments they find more toxic.

In this project, we will use a ranking model called Bert that learns contextual embedding, which has been applied to capture complex query-document relations for search ranking and neural methods, i.e. RNN and CNN.

2.Literature Review

Early literature in automatic online abusive speech detection often involves feature-based classical machine learning techniques such as Support Vector Machine. Simple surface features such as Bag of Words, token and character n-grams are heavily used. Linguistic features such as length of comment in tokens, average length of word, and number of punctuations are often used to explicitly look for inflammatory words (J.-M. Xu et. Al, 2012).

In the earlier version of this competition where Classification of toxic comments was the challenge, one of the teams implemented RNN specifically long-short term memory (LSTM) units, gated recurrent units (GRU), and convolutional neural networks (CNN) to label comments as toxic, severely toxic, hateful, insulting, obscene, and/or threatening. Each of their models reached **over** 88.4% accuracy on cross-validation and the held-out test set. They also stated that However, as the competition and the results of other participants showed it is worth paying attention to a relatively new approach to NLP such as ULMFiT (Howard and Ruder 2018), BERT and ELMo language models, because they gave as good if not better results. These models can also be trained on a generic body, for example, the Wikipedia or the twitter comments.

3. Implementation

3.1 Dataset

The Conversation AI team, a research initiative founded by Jigsaw and Google worked on tools to help improve online conversation. One area of focus was the study of negative online behaviors, like toxic comments such as comments that are rude, disrespectful or otherwise likely to make someone leave a discussion. They built a range of publicly available models served through the Perspective API, including toxicity.

The training data for this competition will be used from previous Jigsaw competition such as Toxic Comment Classification Challenge, Jigsaw Unintended Bias in Toxicity Classification and Jigsaw Multilingual Toxic Comment Classification The original training set has 1804874 examples of

public comments. Additionally, each example is labelled with a variety of identity attributes, representing the identities that are mentioned in the comment. For testing data, a csv file is provided that contains approximately 14 thousand comments on which the task is to predict a score that represents the relative toxic severity of the comment. Comments with a higher degree of toxicity will receive a higher numerical value compared to comments with a lower degree of toxicity; scores are relative, and not constrained to a certain range of values.

In this data set there are 8 columns. They are id, comment text, Toxic, sever toxic, obscene, threat, identity hate, insult

Since these datasets have already been used in previous Kaggle competitions they have been checked for any missing values and replaced with mean value of the column. Once this is done we can now load the data set.

```
In [3]: df = pd.read_csv("train.csv")
print(df.shape)

for col in ['toxic', 'severe_toxic', 'obscene', 'threat', 'insult', 'identity_hate']:
    print(f'***** {col} *****')
    display(df.loc[df[col]==1, ['comment_text', col]].sample(10))
```

(159571, 8)
***** toxic *****

	comment_text	toxic
158984	hi, idiot, why are you delate my talking, just come out say	1
30485	Your account got fucked?	1
53542	you wanna know what I think tony? \n\nand yeh\ningo ahed n label this az lamandrewice\n\nit wil only make me laugh at ur ignorance even mo\n\ninur just a homofobe yeh blatez\n\nin'sup blud?\n\ninshor maytel\n\ninwanna no wot the prob iz with u?\n\ninu need 2 fin get a lyf mayt!\n\ninget a gfl\n\ninget a b...	1
63163	ChrisO proven Criminal and Fucken Homo\n\nChris, you mother fucker...all what you want to know about ChrisO you can find at www.ChrisO.homo.com'Bold text'	1
27702	Right, so an E-4 airman is going to warn off an O-5 LTC who's in the CID. whatever. In any event, that guy is nothing but a vandal. But I don't see you trying to warn him off, kid. In fact, you did nothing.	1
69939	"Rob, is a malicious administrator who likes to use the ""power"" that he has been given here on Wikipedia to nitpick and harass well meaning editors. He is a notorious asshole, and it is only a matter of time before enough people get fed up with his antics and DEMAND that his administratorshi...	1
74085 \n\nyou know you done fucked up, right?	1

Figure 3.1 Loading dataset

3.2 Baseline Model Bert

BERT is Bidirectional Encoder Representations from Transformers. It is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks.

BERT also is pre-trained on a large set of unlabelled text including the entire Wikipedia.

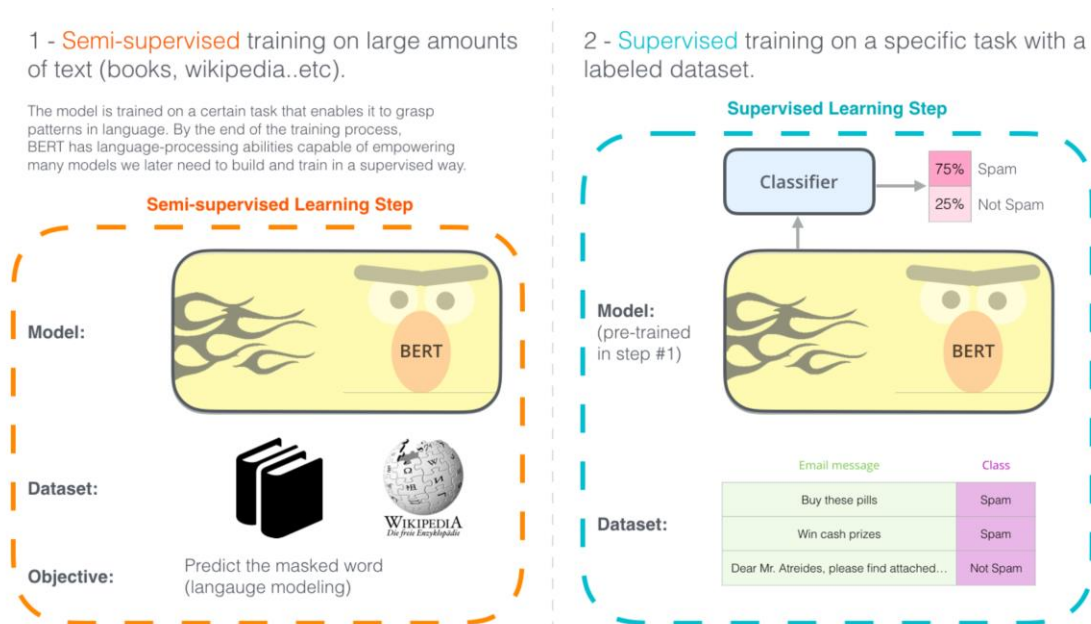


Figure 3.2 Two steps of how Bert is developed

3.3 Model Architecture

Both BERT model sizes have a large number of encoder layers (which the paper calls Transformer Blocks) — twelve for the Base version, and twenty four for the Large version. These also have larger feedforward-networks (768 and 1024 hidden units respectively), and more attention heads (12 and 16 respectively) than the default configuration in the reference implementation of the Transformer in the initial paper (6 encoder layers, 512 hidden units, and 8 attention heads).

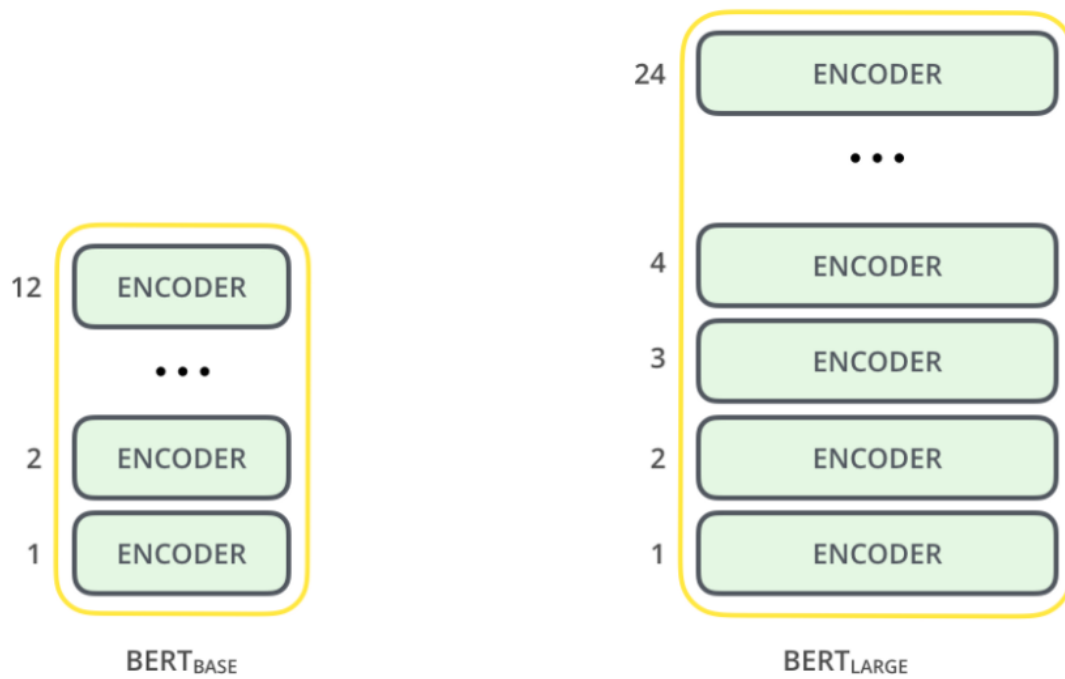


Figure 3.3 Bert Architecture

3.4 Text Preprocessing

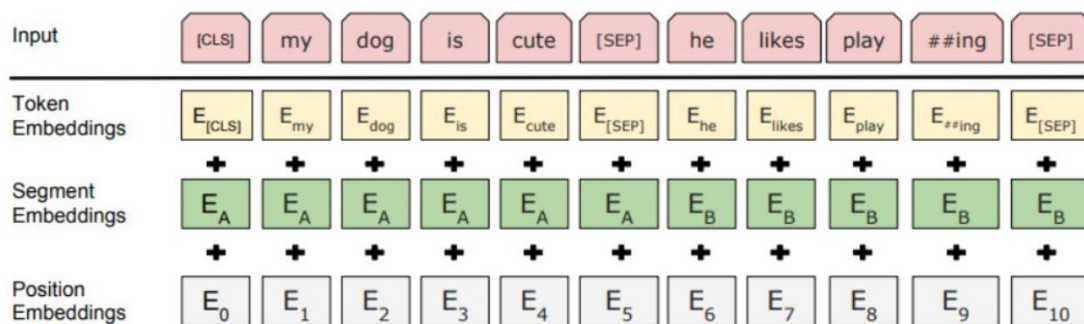


Figure 3.4 BERT text processing

Position Embeddings: BERT learns and uses position embeddings to represent the position of words in a sentence. These are added to overcome Transformer's limitations of not being able to capture sequence or order information, unlike RNNs.

Segment Embeddings: BERT can also take sentence pairs as inputs for tasks (Question-Answering). That's why it learns a unique embedding for the first and the second sentences to help the model distinguish between them. In the above example, all the tokens marked as EA belong to sentence A (and similarly for EB)

Token Embeddings: These are the embeddings learned for the specific token from the WordPiece token vocabulary

3.5 RoBERTa Model

This model was proposed in 2019 by Yinhan Liu Et. al. in RoBERTa: A Robustly Optimized BERT Pretraining Approach

RoBERTa stands for Robustly Optimized Bidirectional Encoder Representations from Transformers. It builds on BERT and modifies key hyperparameters, removing the next-sentence pretraining objective and training with much larger mini-batches and learning rates. It has the same architecture as BERT, but uses a byte-level BPE as a tokenizer and uses a different pretraining scheme (Liu, et al., 2019).

RoBERTa doesn't have `token_type_ids`, so it is not necessary to indicate which token belongs to which segment. It is an extension of BERT with changes to the pretraining procedure. The modifications include:

- Training the model longer, with bigger batches, over more data
- Removing the next sentence prediction objective
- Training on longer sequences
- Dynamically changing the masking pattern applied to the training data. The authors also collect a large new dataset of comparable size to other privately used datasets, to better control for training set size effects.

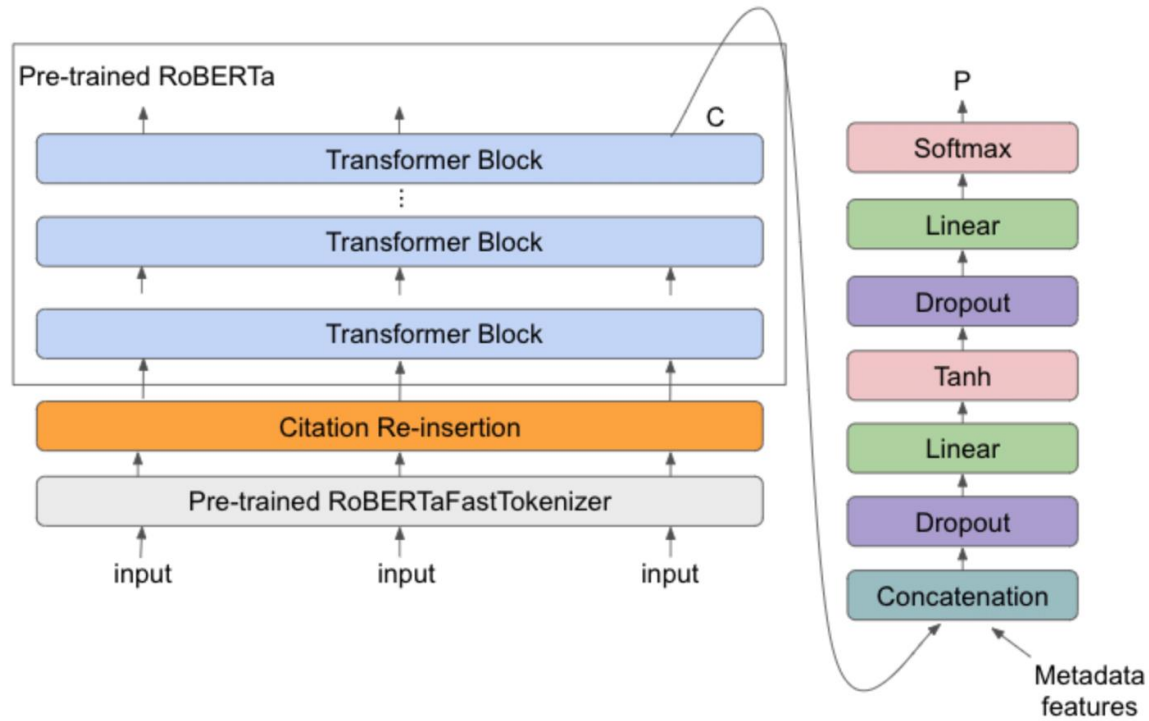


Figure 3.5 RoBERTa Model

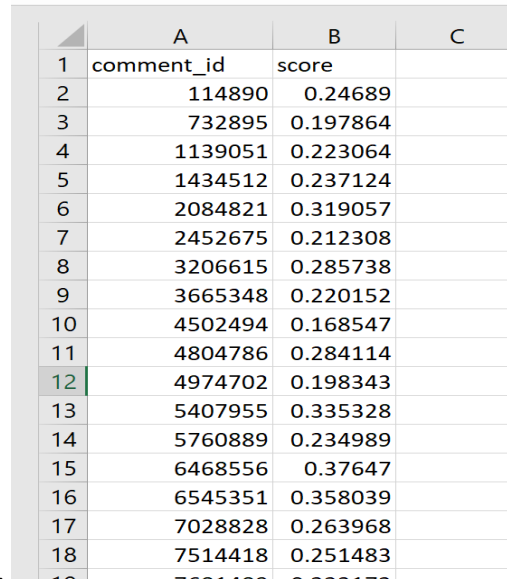
3.6 Model Training

For all our experiments we have used pre-trained RoBERTa model for our dataset. First the dataset set was split into 3 sections, training, testing and validation. We have also exploited K-Folds cross-validator. This Provides train/test indices to split data in train/test sets. Split dataset into k consecutive folds. Each fold is then used once as a validation while the k - 1 remaining folds form the training set. The learning rate was kept between [.1 and .001]. We trained our model with batch size of 64 as the training time per epoch was faster than smaller batch sizes. The loss function used was:

$$\text{loss}(x_1, x_2, y) = \max(0, -y * (x_1 - x_2) + \text{margin})$$

4. Results

The proposed RobBERTa architecture was able to rank comments based on their probability of severity. First once the comments are given a probability score they are saved in an excel sheet as shown in figure 4.1



	A	B	C
1	comment_id	score	
2	114890	0.24689	
3	732895	0.197864	
4	1139051	0.223064	
5	1434512	0.237124	
6	2084821	0.319057	
7	2452675	0.212308	
8	3206615	0.285738	
9	3665348	0.220152	
10	4502494	0.168547	
11	4804786	0.284114	
12	4974702	0.198343	
13	5407955	0.335328	
14	5760889	0.234989	
15	6468556	0.37647	
16	6545351	0.358039	
17	7028828	0.263968	
18	7514418	0.251483	

Figure 4.1 scoring comments

The toxic data validation accuracy was 68.28% only after 5 folds. Training had to be reduced due to lack of computational power available as this one trained on google collab. The comments are then ranked among each other to find out which is more toxic between two.

worker		less_toxic	more_toxic	upper_1	upper_2	p1	p2	diff
27415	730	"\n\nRemoval of section\n\nHowever a removal of content is considered, a large chronology spanning a great length is poor way to put together an article. Please don't be an asshole when writing your edit summaries. A stock {{test}} message in response to a good-faith edit is also rather obnoxio...	Werd to ya \n\nActually, the version you push has not been agreed upon, quit pushing it. Also, everytime I masturbate, I think of you. Go ahead and delete this comment, but if you were true to the cause, you would leave it.	0.015251	0.026786	0.016925	0.016923	0.000002
10004	254	Sockpuppet \n\nI would like you to tell everyone I'm not a sockpuppet of you, just to make sure everyone is clear.	Sheodred? Didn't you also accuse these other IPs: 58.7.244.17 , 93.107.209.165 , 109.77.50.2 , 143.239.103.219 of being her/him? lol you poor sad little man 143.239.7.4	0.026316	0.023392	0.017967	0.017962	0.000005
10003	728	Sockpuppet \n\nI would like you to tell everyone I'm not a sockpuppet of you, just to make sure everyone is clear.	Sheodred? Didn't you also accuse these other IPs: 58.7.244.17 , 93.107.209.165 , 109.77.50.2 , 143.239.103.219 of being her/him? lol you poor sad little man 143.239.7.4	0.026316	0.023392	0.017967	0.017962	0.000005
24732	716	You could have unlocked it - as you see below the reason for the	I never said you weren't allowed to blank your talk page, so what is the reason for this desperate attempt at a sermon? I merely stated that it's cute how you do, but I don't blame you. As a troll and	0.039062	0.015106	0.014972	0.014966	0.000006

Figure 4.2 Finding more toxic comments by comparison

Training and validation loss for the dataset was recorded for each fold Figure 4.3 shows the overall loss for the dataset.

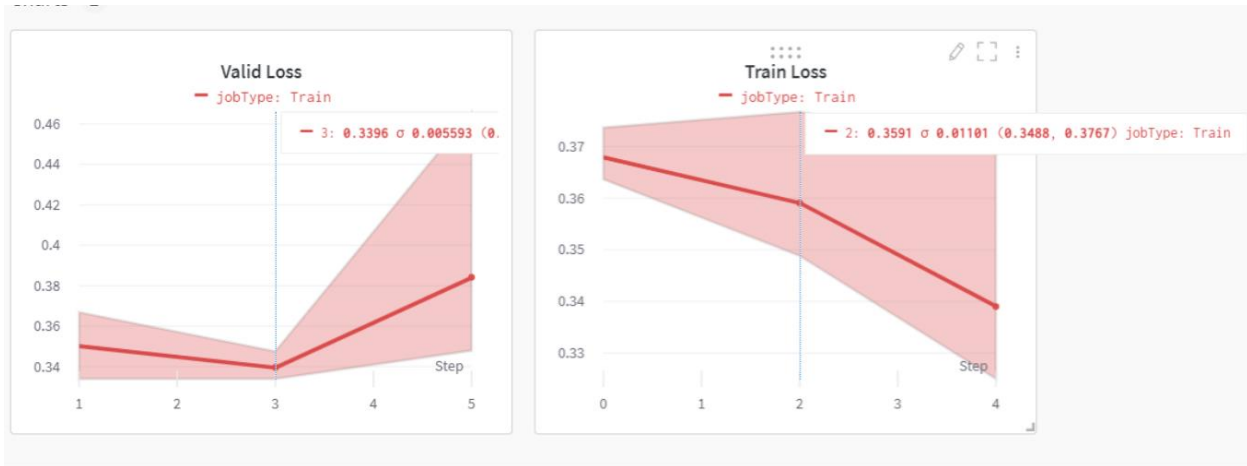


Figure 4.3 Training and Validation Loss for Kaggle Rate Severity of Toxic Comment Dataset

5. Conclusion and Future works

In this project, we have proposed to rank severity of toxic comments. we showed that with minimal preprocessing techniques we are able to achieve a good model performance. We show the effectiveness of our proposed model against strong benchmark algorithms and that it outperforms others. In this work, we did basic preprocessing of the data. However, in future we intend to explore more preprocessing techniques for the dataset, like data augmentation using translation approaches and methods to deal with misspelled words. We can also train for larger epochs as this was not done this time due to limitations in Computing power. Training for longer duration will improve the overall efficiency of the model and reach the optimum accuracy that the model can predict. For future works we can incorporate pretrained BERT and ELMo as the embedding layer, and training separate models for ranking each toxic comment.

6. References

- [1] Kwak, H.; Blackburn, J.; Han, S. Exploring cyberbullying and other toxic behavior in team competition online games. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Korea, 18–23 April 2015; pp. 3739–3748.
- [2] D. Yin, B. D. Davison, Z. Xue, L. Hong, A. Kontostathis, and L. Edwards, “Detection of harassment on web 2.0,” 2009.
- [3] J.-M. Xu, K.-S. Jun, X. Zhu, and A. Bellmore, “Learning from bullying traces in social media,” in Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT ’12, (Stroudsburg, PA, USA), pp. 656–666, Association for Computational Linguistics, 2012.
- [4] Liu, Y., Ott, M., Goyal, N. & Du, J., 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *Paul G. Allen School of Computer Science & Engineering*,.