Comparison of Language Models on Classification of Toxic Comments Subject Matter

Arlan Kalin
20192013 / UNIST
arlan914unist
@unist.ac.kr

Almas Dossay
20202016 / UNIST
almasdossay
@unist.ac.kr

Artem Kim
20202021 / UNIST
kim21artem
@unist.ac.kr

Abstract

This research paper presents an in-depth comparison of two widely recognized models, FastText and BERT, for the purpose of classifying toxic comments. FastText is known for its efficiency in handling large volumes of text data, making it an ideal choice for analyzing online comments. On the other hand, BERT excels in understanding the contextual meaning of words, which enhances its ability to accurately identify toxic comments. The study aims to evaluate the performance of these models by considering their strengths and limitations, utilizing precision, recall, accuracy, and AUC-ROC metrics. The results indicate that both FastText and BERT demonstrate strong performance, with BERT showing superior effectiveness in distinguishing between different types of toxic comments. This study provides valuable insights into the unique advantages of each model and highlights the potential of transformer-based models, specifically BERT, for complex text classification tasks, particularly in the area of toxic comment identification.

1 Introduction

Classifying toxic comments is crucial for online comment moderation. This research compares FastText and BERT models to assess their performance in this task. FastText is chosen for efficiency with large-scale data, while BERT is known for contextual understanding. The goal is to understand how their strengths and limitations impact their effectiveness in toxic comment classification. FastText constructs word vectors using character n-grams, enabling it to handle complex languages and understand word meanings. Data preprocessing includes tokenization and stop-word removal. FastText is trained using negative sampling. BERT is a transformer-based model that excels in pre-training for natural language processing. It understands word context from both directions, improving understanding of language. The pre-trained BERT model is adapted for sequence classification, utilizing tokenization and transformer layers. Classification is done using the final hidden state of the '[CLS]' token. Model fine-tuning employs the Binary Cross Entropy with Logits loss function and the Adam optimizer. The research focuses on multi-label classification, where comments can belong to multiple classes. Both models use a sigmoid activation function for class probability transformation. Evaluation includes precision, recall, accuracy, and AUC-ROC. Results indicate good performance for both models, with BERT being better at distinguishing between classes. In conclusion, this study provides valuable insights into the strengths of FastText and BERT, highlighting the potential of transformer-based models like BERT for complex text classification, particularly in toxic comment classification.

2 Related Literature

Social networks, such as Twitter, have been extensively studied for sentiment analysis. One notable paper focused on sentiment analysis is 'Sentiment Analysis of Twitter Data'[1]. This literature review explores the key aspects of that paper and draws parallels with our project, which aims to identify toxic content.

The reviewed paper proposed a sentiment analysis model for Twitter using BERT. It outlined a systematic approach involving dataset cleaning, preprocessing, BERT tokenizer encoding, and classification into five sentiment indexes. The fine-tuned BERT model achieved state-of-the-art results in sentiment classification.

Inspired by this literature, our project focuses on identifying toxic comments. We conducted a comparative study of FastText and BERT models. FastText, known for handling large-scale text data efficiently, was used for processing a substantial corpus of online comments. BERT, on the other hand, excels at capturing contextual word information crucial for accurate toxic comment identification.

In our project, we followed a similar data preprocessing approach to the reviewed paper. Tokenization, stop word removal, and representing comments as averages of word vectors were applied in FastText. For BERT, we used the pretrained model, added a linear layer, and processed tokenized comments using transformer layers.

A significant challenge we addressed was multilabel classification, as comments can belong to multiple classes. To handle this, we incorporated a sigmoid activation function in the final layer of both FastText and BERT models. This provided confidence values for each class prediction.

Evaluation metrics such as precision, recall, accuracy, and AUC-ROC were used to assess model performance. The FastText model showed high precision and recall in predicting toxic comments but had limitations in distinguishing between positive and negative classes. Conversely, the BERT model demonstrated exceptional accuracy and a high AUC-ROC score, indicating its efficacy in classifying toxic and non-toxic comments.

In summary, our project was motivated by the reviewed literature on Twitter sentiment analysis using BERT. We focused on identifying toxic comments and conducted a comparative study of FastText and BERT models. Our findings highlighted the advantages of transformer-based models, particularly BERT, in complex text classification tasks. By leveraging insights from the literature, our project contributes to the ongoing research on toxic content identification in social media platforms.

3 Data

The Jigsaw Toxic Comment Classification Dataset is a collection of comments from Wikipedia. The dataset contains 8 variables: id, comment text, toxic, severe toxic, obscene, threat, insult, and identity hate.
The "id" variable represents a unique identifier for each comment in the dataset. The "comment text" variable contains the actual text of the comments in string format. The remaining variables, namely "toxic," "severe toxic," "obscene," "threat," "insult," and "identity hate," are binary variables that indicate whether a comment falls into the respective category. A value of 1 represents that the comment is classified as belonging to that category, while 0 indicates the opposite.
The dataset consists of 223,549 comments. Among these comments, the percentages of those classified as toxic, severe toxic, obscene, threat, insult, and identity hate are as follows:
Toxic: 9.5657 percent
Severe Toxic: 0.8777 percent
Obscene: 5.4306 percent
Threat: 0.3082 percent
Insult: 5.0566 percent
Identity Hate: 0.9470 percent
These percentages provide an understanding of the prevalence of different types of toxicity in the dataset. It is important to note that these percentages are specific to the Jigsaw Toxic Comment Classification Dataset and may not be representative of toxicity levels in other contexts or datasets.

4 Preprocessing

For preprocesing, we used regular expressions to eliminate certain patterns from each comment. Here is a breakdown of the patterns:
1)[zero or more characters];
This pattern remove all objects within brackets [].
2) with or without (http), ://, or www;
This pattern remove all objects that are url or website.
3)<zero or more characters> >(one or more occurrence);
This pattern remove all objects within brackets < > or contrains one of more of >.
4) any punctuation;
This pattern remove any punctuation.
5) any new line;
This pattern remove any new line. All words are written in one line.
6) any unregular combinations of words and digits.
This pattern remove any unregular combinations of words and digits. For example, gda45adsf.
This code that we used for our preprocesing.

5 Model

In this research, we aimed to perform a comparative study of two influential models, **FastText**[3] and **BERT**[2] (Bidirectional Encoder Representations from Transformers), for the task of toxic comment classification.

The primary motivation behind choosing these models lies in their unique strengths and wide usage in the realm of natural language processing (NLP). FastText is known for its efficiency and its ability to handle large-scale text data, making it a

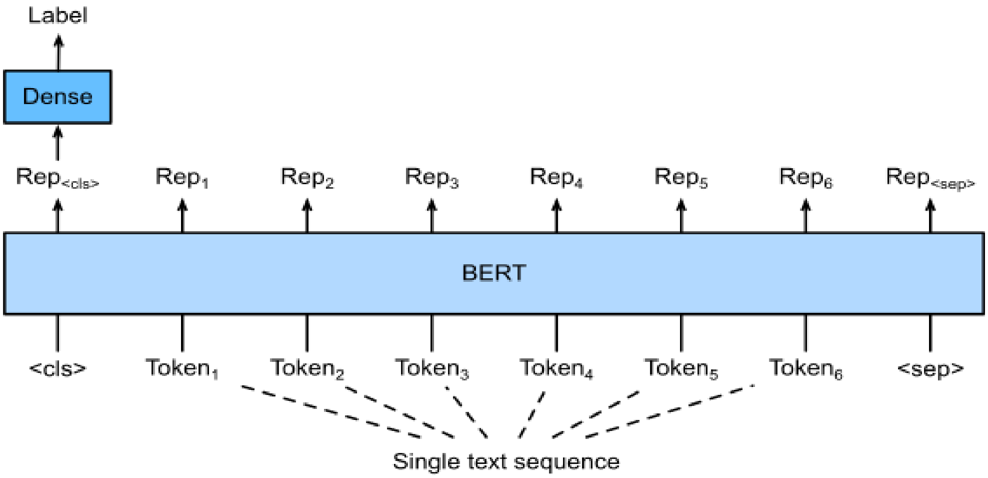


Figure 1: The general Architecture of BERT for Sequence Classification Model.

suitable choice for a text classification task such as ours, which involves dealing with a large corpus of online comments. On the other hand, BERT is celebrated for its ability to grasp the context of words in a sentence, offering a nuanced understanding of language that has the potential to increase accuracy in identifying toxic comments.

By employing and comparing these two models, we aim to analyze their performances in classifying toxic comments and investigate how their respective strengths and limitations influence their efficacy in this particular task. It is anticipated that this comparative study will offer valuable insights into the use of FastText and BERT for text classification tasks, specifically in the area of online comment moderation and toxicity identification.

5.1 FastText

FastText differentiates itself through its use of subword information, implementing character n-grams to construct word vectors. This process not only considers the word as a single entity but also incorporates the different n-grams within the word. For instance, using trigrams (n=3) on the word ’example’, FastText processes individual character trigrams: ’exa’, ’xam’, ’amp’, ’mpl’, ’ple’, and so forth. These, coupled with the whole word, contribute to the final word vector. This approach enables FastText to manage morphologically complex languages and comprehend semantic word meanings, even with out-of-vocabulary (OOV) words. For our research, we first preprocessed the data, which included tokenization and stop words removal. Then, each comment was represented as an average of its word vectors, including the n-gram representations, creating a dense representation for each comment. FastText was then trained on these vectors using a negative sampling approach for optimization.

5.2 BERT

BERT is a famous transformer-based machine learning technique used for pre-training natural language processing (NLP) tasks. It consists of multiple layers of transformer encoders stacked on top of each other. For instance, the ’bert-base-uncased’ model, which we used in our research, consists of 12 transformer layers, each with 12 self-attention heads, and has an embedding size of 768, resulting in a total of approximately 110 million parameters.

One of the defining characteristics of BERT is its bidirectional nature. Unlike traditional language models that predict subsequent words based on preceding ones, BERT is capable of understanding the context of a word based on all of its surroundings (left and right of the word). This is achieved through a mechanism called Transformer, which uses self-attention heads to weigh the significance of words in the sentence. This ability to understand word context from both directions enhances the model’s understanding of the language semantics and syntax.

In our research, we used the pre-trained BERT model as a starting point, which means that the model had already learned a lot about the English language from a large corpus of text (Wikipedia and BooksCorpus). This pre-training phase makes BERT a powerful tool as it brings in substantial prior knowledge before being fine-tuned on specific tasks like ours.

We adapted BERT for our task of sequence classification by adding a single linear layer on top for classification. The comments in our dataset were tokenized using the BERT tokenizer, which also provided the necessary segment and positional embeddings. Each comment was then input into the model. The transformer layers processed the tokenized comments, and the final hidden state of the '[CLS]' token was used to classify the comment. This '[CLS]' token captures the contextual representations of the entire sequence, making it suitable for classification tasks.

The model was trained using the Binary Cross Entropy with Logits loss function and the Adam optimizer. We selected a learning rate of 2e-5 and trained the model for four epochs. To convert the outputs of the model to probabilities, a sigmoid activation function was applied to the final layer, which effectively maps the output to a value between 0 and 1. This value represents the model's confidence that the comment belongs to a particular class.

5.3 Multi-label Classification

Our task is a multi-label classification problem, meaning each comment could potentially belong to more than one of the six classes. To handle this, we used a sigmoid activation function in the final layer of both models. This function transformed the output for each class into a value between 0 and 1, representing the model's confidence that the comment belongs to that class.

6 Training

We trained two models, fastText and BERT, for classification purposes. The fastText model is based on the Continuous Bag of Words (CBOW) approach and utilizes tokens on both the word and sub-word levels. It creates a dictionary from the training dataset to classify unseen data. To ensure a balanced training process, we used a dataset with a 50/50 ratio of toxic and non-toxic comments. This balanced distribution helps the model learn to handle both types of comments effectively.

For the BERT model without cross-validation, we used a separate validation set to evaluate its performance. The validation set contained labeled data that the model had not seen during training. As the BERT model was trained over multiple epochs, we saved the model with the best results based on the error rate obtained from the validation set. This saved model represents the best-performing state of the model throughout the training process.

In the case of BERT with cross-validation, the training dataset is divided into five parts or folds. Each part was used as a validation set while the remaining subsets were used for training in different iterations. By using cross-validation, you obtain a more robust estimation of the model's performance by considering multiple validation sets. The performance of the BERT model was evaluated by averaging the best results based on the error rate obtained from each validation set. This approach provides a more comprehensive understanding of the model's performance across different subsets of the data.

7 Results

In this section, we discuss the evaluation process for the performance of our FastText and BERT models. We employed metrics such as precision, recall, accuracy, and the Area Under the Receiver Operating Characteristics curve (AUC-ROC) to evaluate the models' performance.

7.1 FastText Model Evaluation

For the FastText model, we evaluated its performance using precision, recall, and AUC-ROC.

Precision and recall are crucial metrics for assessing classification models' performance. Precision measures the number of correctly predicted positive instances out of the total predicted positive instances, indicating the model's ability to accurately predict positive (toxic comment) classes. Recall (also known as sensitivity or true positive rate) measures the ability of the model to correctly identify actual positive instances.

The FastText model achieved a validation precision and recall of **95.76%**, indicating a high degree of reliability in predicting the positive (toxic comment) class.

However, the AUC-ROC score of the FastText model was considerably lower at **63.4%**. The AUC-ROC is an essential metric as it illustrates the model's capability to distinguish between the positive and negative classes across various classification thresholds. A perfect model achieves an AUC of 1, while a model no better than random chance achieves an AUC of 0.5. The relatively lower AUC-ROC score implies that, although the FastText model is effective in predicting the positive class, it is less robust in distinguishing between the positive and negative classes across various thresholds compared to the BERT model.

7.2 BERT Model Evaluation

For the BERT model, we utilized a strategy of **5-fold cross-validation** for evaluation to ensure a more reliable and generalized performance. This method divides the dataset into 5 subsets. The model is trained 5 times, each time using a different subset for validation and the remaining subsets for training.

Accuracy. Accuracy, a key evaluation metric, measures the proportion of correct predictions made by the model. We calculated the accuracy of the BERT model by comparing the model's predictions against the actual labels in the training and validation datasets. The BERT model, following cross-validation, achieved a training accuracy of **98.8%** and validation accuracy of **97.8%**, indicating exceptional performance in correctly predicting the classes of comments in both the training and validation data.

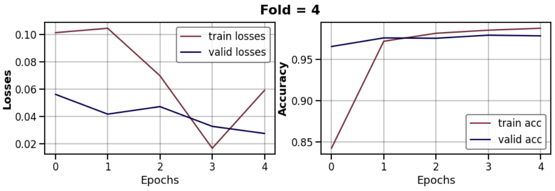


Figure 2: Content image.

AUC-ROC. We calculated the AUC-ROC for the BERT model using the roc_curve function from the sklearn.metrics module and the actual labels and predicted probabilities from the validation dataset.

The BERT model achieved an AUC-ROC score of **99.19**, indicating that the model shows high separability and can effectively distinguish between toxic and non-toxic comments.

8 Discussions

8.1 Discussion of results

In this study, we compared the performance of FastText and BERT models for toxic comment classification. FastText is known for its efficiency in handling large-scale text data, while BERT excels in understanding contextual language. Our aim was to assess their effectiveness in identifying toxic comments and understand their strengths.

FastText utilizes character n-grams to construct word vectors, enabling it to handle complex languages and capture word meanings, even for unfamiliar words. We preprocessed the data by tokenizing and removing stop words. Each comment was represented as an average of its word vectors, including n-gram representations. FastText achieved a validation precision and recall of **95.76%**, indicating its reliability in accurately predicting toxic comments.

On the other hand, BERT employs a transformer-based approach to comprehend word context bidirectionally, enhancing its understanding of language semantics. Using a pre-trained BERT model, we adapted it for sequence classification and trained it with Binary Cross Entropy loss. BERT achieved exceptional performance with a training accuracy of **98.8%** and a validation accuracy of **97.8%**. It demonstrated an AUC-ROC score of 99.19, highlighting its effectiveness in distinguishing between toxic and non-toxic comments.

Overall, both models performed well, but BERT showed superior performance in accurately classifying toxic comments. Its bidirectional understanding of language and contextual comprehension contributed to its outstanding results.

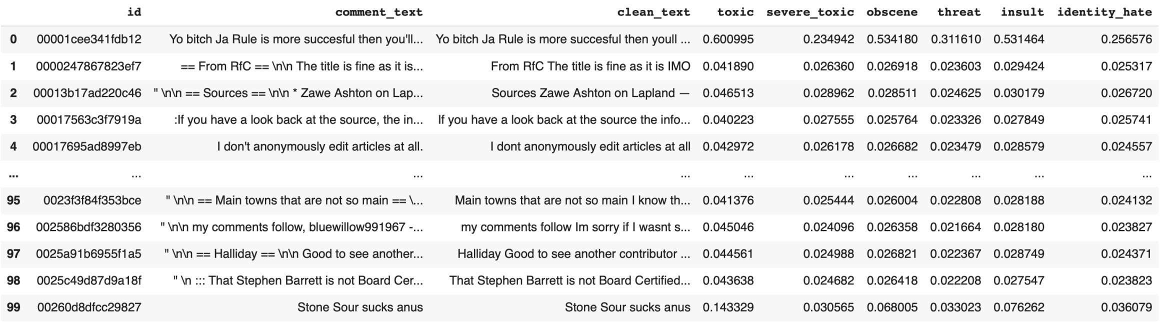
8.2 Future plans

In future plans, we aim to further optimize the models for enhanced classification accuracy. We also intend to apply these models to multilingual platforms and international movies for toxicity identification. Additionally, implementing them on major social networks like TikTok, Instagram, and Twitch would aid in content moderation and create safer online environments.

This research provides valuable insights into the advantages of FastText and BERT models for toxic comment classification. By refining and expanding their capabilities, we can make a significant impact in combating online toxicity.

Conclusion

In conclusion, both models performed well on the toxic comment classification task, but the BERT model displayed superior performance, particularly in distinguishing between classes across different classification thresholds. This comparative study provides valuable insights into the advantages of each model, highlighting the potential of



	id	comment_text	clean_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	00001cee341fdb12	Yo bitch Ja Rule is more succesful then you'll...	Yo bitch Ja Rule is more succesful then youll ...	0.600995	0.234942	0.534180	0.311610	0.531464	0.256576
1	0000247867823ef7	== From RfC == \n\n The title is fine as it is...	From RfC The title is fine as it is IMO	0.041890	0.026360	0.026918	0.023603	0.029424	0.025317
2	00013b17ad220c46	" \n\n == Sources == \n\n * Zawe Ashton on Lap...	Sources Zawe Ashton on Lapland —	0.046513	0.028962	0.028511	0.024625	0.030179	0.026720
3	00017563c3f7919a	:If you have a look back at the source, the in...	If you have a look back at the source the info...	0.040223	0.027555	0.025764	0.023326	0.027849	0.025741
4	00017695ad8997eb	I don't anonymously edit articles at all.	I dont anonymously edit articles at all	0.042972	0.026178	0.026682	0.023479	0.028579	0.024557
...	...	...	...	...	...	...	...	...	...
95	0023f3f84f353bce	" \n\n == Main towns that are not so main == \...	Main towns that are not so main I know th...	0.041376	0.025444	0.026004	0.022808	0.028188	0.024132
96	002586bdf3280356	" \n\n my comments follow, bluewillow991967 -...	my comments follow Im sorry if I wasnt s...	0.045046	0.024096	0.026358	0.021664	0.028180	0.023827
97	0025a91b6955f1a5	" \n\n == Halliday == \n\n Good to see another...	Halliday Good to see another contributor ...	0.044561	0.024988	0.026821	0.022367	0.028749	0.024371
98	0025c49d87d9a18f	" \n ::: That Stephen Barrett is not Board Cer...	That Stephen Barrett is not Board Certified...	0.043638	0.024682	0.026418	0.022208	0.027547	0.023823
99	00260d8dfcc29827	Stone Sour sucks anus	Stone Sour sucks anus	0.143329	0.030565	0.068005	0.033023	0.076262	0.036079

Figure 3: Representation of the model's confidence that the comment belongs to that class

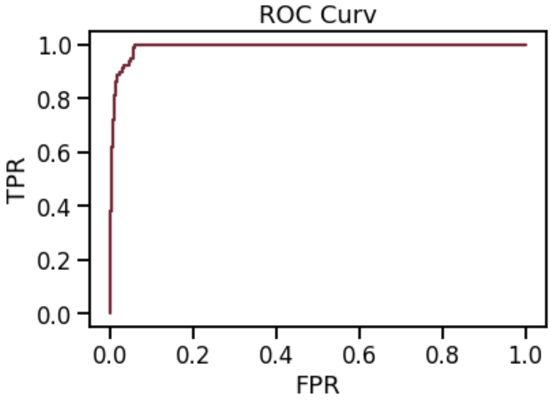


Figure 4: ROC-Curve for Cross Validation of BERT.

transformer-based models such as BERT for complex text classification tasks.

References

[1] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca J Passonneau. Sentiment analysis of twitter data. In *Proceedings of the workshop on language in social media (LSM 2011)*, pages 30–38, 2011.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.