Evaluating The Effectiveness of Capsule Neural Network in Toxic Comment Classification using Pre-trained BERT Embeddings

Md Habibur Rahman Sifat[1], Noor Hossain Nuri Sabab[2] and Tashin Ahmed[3]

Abstract—**Large language models (LLMs) have attracted considerable interest in the fields of natural language understanding (NLU) and natural language generation (NLG) since their introduction. In contrast, the legacy of Capsule Neural Networks (CapsNet) appears to have been largely forgotten amidst all of this excitement. This project's objective is to reignite interest in CapsNet by reopening the previously closed studies and conducting a new research into CapsNet's potential. We present a study where CapsNet is used to classify toxic text by leveraging pre-trained BERT embeddings (bert-base-uncased) on a large multilingual dataset. In this experiment, CapsNet was tasked with categorizing toxic text. By comparing the performance of CapsNet to that of other architectures, such as DistilBERT, Vanilla Neural Networks (VNN), and Convolutional Neural Networks (CNN), we were able to achieve an accuracy of 90.44%. This result highlights the benefits of CapsNet over text data and suggests new ways to enhance their performance so that it is comparable to DistilBERT and other reduced architectures.**

I. Introduction

In this age of cutting-edge technology, internet access is literally available to everyone. As it becomes more accessible, the cyber world becomes more susceptible to vulgar and abusive individuals. Therefore, it is becoming necessary for social media platforms to analyze and investigate inappropriate content on their platform. As more people join every day, it becomes increasingly difficult to manually police the Internet. Thus, an automated system that can detect toxic messages, quantify them, and then take the necessary actions is required.

This study aims to evaluate CapsNet against alternative neural network architectures for detecting potential toxicity in text data. We collected a dataset from Jigsaw (a Google think tank) via a 2020 Kaggle competition[a]. As performance evaluation of CapsNet is our primary concern, we have utilized googletrans library to translate non-English texts from the dataset and have not yet considered the issue from a multilingual perspective.

We implemented toxic text classification using CapsNet with pre-trained BERT embeddings (bert-base-uncased) and worked on a rich multilingual dataset. We worked on CapsNet to overcome CNN's subpar performance and achieved an accuracy of 90.44%, demonstrating CapsNet's performance over text data. Lastly, we implemented DistilBERT utilizing the same pretrained bert tokenizer as other methods (VNN and CNN) to achieve the highest accuracy of 93.24%. Our long-term objective is to make CapsNet's performance comparable to that of DistilBERT and other reduced architectures.

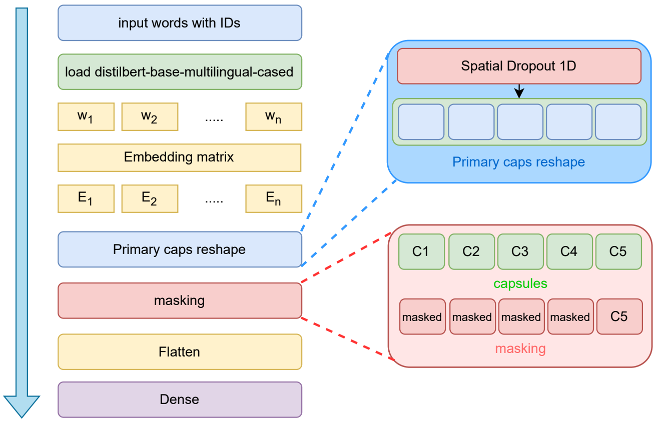

Fig. 1. Presented CapsNet architecture. The model takes word IDs as input and employs pre-trained BERT embeddings to extract context from text. A spatial dropout layer is applied to the BERT embeddings to prevent overfitting. The capsule layer receives the modified embeddings and learns to represent the input text as a collection of capsules, where each capsule represents a particular characteristic or attribute of the text. The capsule outputs are then fed into dense layers in order to learn higher-level text representations. The final dense layer generates the output prediction, which indicates the classification or label of the input text.

We implemented a number of neural network architectures, including VNN, CNN, CapsNet, and DistilBERT. We demonstrated that CapsNet extracts more features and performs better than CNN on text data.

II. Related Works

Classification of sentiment regarding toxicity has been massively studied in the recent years. Specially, researchers applied various machine learning algorithms on the social media data to identify the toxicity in online platform. Naguyen and Naguyen [8] proposed a DeepCNN and Bidirectional LSTM combined model which produce sentence wide feature embeddings from word level embeddings and achieved a good accuracy of 86.63%. Chu and June [2] achieved 93% accuracy on comment abuse classification problem using word embeddings by RNN with LSTM, CNN and character embeddings by CNN. Khieu and Narwal [5] showed CNN and LSTM based model which works for character level but did not works for sentence level.

Xie et al [10] proposed an ensemble method for the multilingual toxic comment classification where their model achieved an AUC of 0.9485 for the validation set. Some

[1] Md Habibur Rahman Sifat is with Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong. habib.sifat@connect.polyu.hk

[2] Noor Hossain Nuri Sabab is with Department of CSE, United International University, Dhaka, Bangladesh. nsabab@aol.com

[3] Tashin Ahmed is with the AI Team, Smart Studios, B'Kara, Malta. tashin@smartstudios.io

[a] https://www.kaggle.com/competitions/jigsaw-multilingual-toxic-comment-classification

TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON) | 979-8-3503-0219-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/TENCON58879.2023.10322429

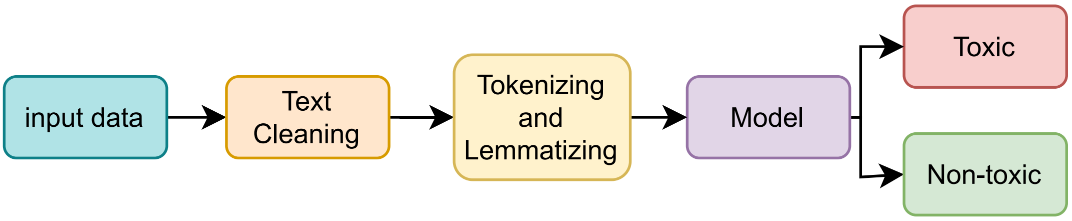

Fig. 2. A straightforward general structure of the experiment that have been performed on the text data.

researchers [1], [6] focused on regional language such as Brazilian Portuguese, Russian. Leite et al [6] achieved 75% of F1-score but using BERT model but it's struggling with data imbalancing and could not detect some comments. Xu et al [11] proposed an ensemble model that combines monolingual and multilingual models to achieve a better performance for both cases using XLM-RoBERTa multilingual fine-tuning model. Song et al [9] proposed a fusion model for the same dataset where they combined multiple loss function and multiple pre-trained model and their accuracy is 94.67.

Mazzia et al [7] designed an efficient capsule network which only require 2% of CapsNet parameters to get state-of-the-art result for three different datasets. They used the self-attention routing mechanism among the active capsules almost a fully connected network with additional branch of attention layer. They performed experiment on MNIST, MultiMNIST and smallNORB datasets and showed the effectiveness of CapsNet. Though all datasets are image based and showed the effectiveness of CapsNet in vision domain, we extend it for the text domain as well.

III. DATASET

The multiclass dataset for classifying toxic comments contained over 223,000 comments, with classes including toxic, severe toxic, obscene, threat, insult, and identity hate. The data was discovered to be extremely unbalanced, with the majority of comments being in English. In addition to analyzing sentiment and polarity with NLTK's Sentiment Intensity Analyzer (SIA), we measured readability with the Flesch reading ease, Automated Readability Index (ARI), and Dale-Chall readability measures. The Flesch reading ease was not found to be a reliable indicator of the toxicity of comments, whereas the ARI and Dale-Chall readability measures provided more insight into the dataset shown in Figure 3.

Negative sentiment is a score between 0 and 1 indicating negative or pessimistic emotions. The higher the score, the more negative the comment. The majority of comments in the dataset had low negativity, indicating that they were not toxic or negative. The most frequent negativity value was approximately 0.04, and few comments had a negativity greater than 0.80. When comparing negativity and toxicity, toxic comments had, on average, significantly more negative sentiment than non-toxic comments. At this point, the majority of non-toxic comments had a negativity value of 0, while toxic comments had the lowest negativity. This indicates that a comment with a negativity score of 0 is most likely harmless.

Positive sentiment is a score between 0 and 1 that represents positive or optimistic emotions. A higher score represents a more favorable comment. The majority of comments contained less positivity. Very few comments had a positivity score greater than 0.80. The most common positivity value was approximately 0.08. Comparing positivity and toxicity, we discovered that the graphs for toxic and non-toxic comments were comparable. This suggests that positivity is not a reliable indicator of comment toxicity.

Neutral sentiment refers to the text's lack of bias or opinion. It is a score between 0 and 1, with a higher score denoting a more neutral or impartial comment. The majority of comments were impartial and neutral. This suggests that comments were generally neutral and devoid of strong opinions. When contrasting neutrality and toxicity, non-toxic comments had, on average, higher neutrality values. At 1, there was an abrupt increase in the probability density of non-toxic comments, while the probability density of toxic comments was significantly lower at the same point. This suggests that comments with scores closer to 1 are more likely to be non-toxic.

Compound sentiment refers to a sentence's overall level of emotion. It is a score between -1 and 1, with higher scores indicating greater emotional intensity. The distribution of sentiment is uniform across the spectrum. It has a high standard deviation and random peaks throughout its range. Figure 4 shows detailed view of them. Some sample data are presented in Table I and a basic statistics of the dataset is shown in Table II.

IV. Methodology

In the initial stage of modeling, the data was cleaned to remove unnecessary elements such as punctuation, timestamps, and user information since it consisted of public comments. The data was then tokenized, which involved splitting the sentences into the smallest possible strings, with the aim of further text cleaning, such as lemmatization or converting the output to a data frame for improved use in a model. The study employed a pre-trained BERT-tokenizer for this purpose, which expedited the progress of the work.

For understanding how the processed data works with a neural network, we applied the BERT embeddings into a basic VNN, with 128 neurons in the Dense layer. We used Rectified Linear Unit (ReLU) as activation function and Binary Cross-entropy as the loss function, with Adam as the optimizer. The network provided us with an accuracy of 80.44%, which showed that the pre-trained BERT embeddings were really effective and also showed why it was considered a state-of-the-art language model.

979-8-3503-0219-6/23/$31.00 ©2023 IEEE

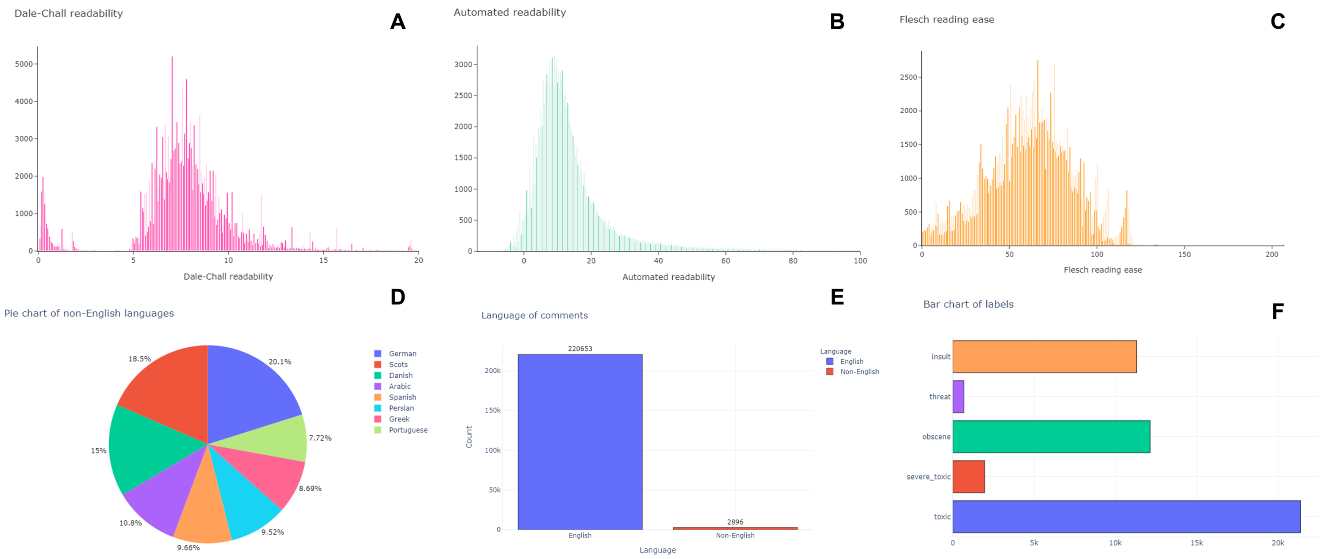

Fig. 3. Metrics to assess the readabilty or ease of understanding texts. A: Dale-Chall readability; B: Automated readability; C: Flesch reading ease. D: Non English language percentages; E: English and non-English language count; F: Toxic class counts.

id	comment_text	toxic	severe toxic	obscene	threat	insult	identity hate
0036621e4c7e10b5	Would you both shut up, you don't run wikipedia, especially a stupid kid.	1	0	0	0	1	0
0021fe88bc4da3e6	My Band Page's deletion. You thought I was gone. Deleting the comment I posted on your 'talk page'...	1	0	1	0	0	0
0007e25b2121310b	Bye! Don't look, come or think of comming back! Tosser.	1	0	0	0	0	0
00472b8e2d38d1ea	A pair of jew-hating weiner nazi schmucks.	1	0	1	0	1	1

TABLE I

SAMPLE DATA FROM THE KAGGLE COMPETITIONS DATASET, UTILISED IN THIS EXPERIMENT. IT INCLUDES A UNIQUE ID FOR EACH ENTRY, LEVEL AND CATEGORY OF TOXICITY, INCLUDING TOXIC, SEVERE TOXIC, OBSCENE, INSULT, AND IDENTITY HATE. 1S AND 0S INDICATE WHETHER THE COMMENT FALLS UNDER THE SPECIFIED CATEGORY.

	toxic	severe_toxic	obscene	threat	insult	identity_hate
0	202165	221587	211409	222860	212245	221432
1	21384	1962	12140	689	11304	2117
mean	0.1	0.01	0.05	0	0.05	0.01
std dev	0.29	0.09	0.23	0.06	0.22	0.1

TABLE II

COUNT OF TOTAL DATA FOR EACH CATEGORY. 1: REPRESENTS THE TOTAL NUMBER OF INSTANCES OF THE SPECIFIED CATEGORY. EACH DATA ENTRY (COMMENT) MAY BELONG TO MULTIPLE CATEGORIES (TABLE I).

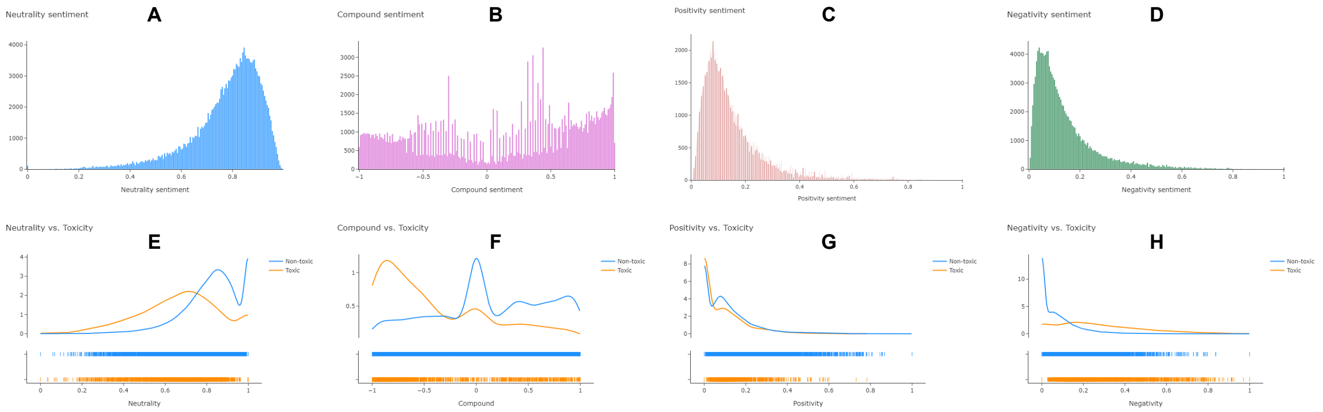

Fig. 4. **Sentiments scores.** A: Neutrality; B: Compound; C: Positivity; D: Negativity. **Comparative analysis against toxicity.** (E: Neutrality, F: Compound, G: Positivity, H: Negativity) vs Toxicity

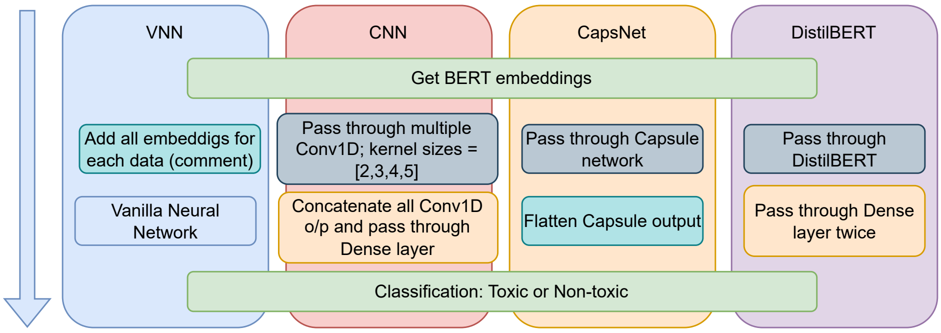
VNN
CNN
CapsNet
DistilBERT
Get BERT embeddings
Add all embeddigs for each data (comment)
Vanilla Neural Network
Pass through multiple Conv1D; kernel sizes = [2,3,4,5]
Concatenate all Conv1D o/p and pass through Dense layer
Pass through Capsule network
Flatten Capsule output
Pass through DistilBERT
Pass through Dense layer twice
Classification: Toxic or Non-toxic

Fig. 5. Basic architecture of all the models (VNN, CNN, CapsNet, DistilBERT) that tested till now. Common layer for every architecture are the BERT embeddings and classifier.

Models	Accuracy	Loss	Val Accuracy	Val Loss
Vanilla Neural Network	85.98%	1.29%	80.44%	1.85%
Convolutional Neural Network	89.77%	0.95%	86.62%	1.03%
CapsNet	93.09%	0.80%	**90.44%**	1.45%
DistilBERT	97.54%	0.58%	**93.24%**	1.74%

TABLE III
MODELS PERFORMANCES IN TERMS OF 5 FOLD CROSS VALIDATION. 90.44% OF CV ACCURACY HAS BEEN ACHIEVED BY CAPSNET. AS A PRE-TRAINED MODEL DISTILBERT ACQUIRED 93.24%.

CNNs are a type of neural network generally used for image recognition problems, although the one-dimensional (1D) version of CNNs can also be used for text-related problems. CNN involves a process called convolution, which is a rather simple algorithm that involves a kernel (a 2D matrix) moving over the entire matrix (word embeddings), calculating dot products with each window along the way. In text classification, a 1D variant of convolution is used where the kernel moves in only one dimension. We used the pre-trained BERT embeddings as input, passed the embeddings through convolutional layers, and got the probability of the comment being toxic. We used four Conv1D layers with 64 neurons in each layer, passed those layers into four max-pool layers, then concatenated the max-pool layers to pass into a Dense layer of 64 neurons. As before, we used ReLU, binary cross-entropy, and Adam for the neural network architecture. We found out that even with a more complex model than the previous one, our accuracy improved very little, with an accuracy of 86.62% from CNN. For test cases, it was seen that it correctly classified all the four inputs that the VNN failed to do, proving that CNN understands spatial relationships better.

Since CNN couldn't provide a significant improvement in accuracy, we designed a CapsNet. CNNs can extract features, but they cannot understand the spatial and proportional relationships between objects in the image. CapsNet solves this problem by understanding the spatial relationships between words (in text) by encoding additional information. A lot of the intuition behind how and why CapsNets work is linked to image recognition, but it can also be used for text-based problems. We used a capsule layer with 5 capsule numbers and 5 dimensions for the model. The layer was then passed through Flatten to change the shape of the data dimension to correctly pass through the Dense layer, with 128 neurons, and same organization as the other models.

Our CapsNet architecture has been built with a BERT-based embedding layer as the underlying structure for text classification tasks. The model accepts word IDs as input and employs pre-trained BERT embeddings to extract context from text. A spatial dropout layer is applied to the BERT embeddings to prevent overfitting. The capsule layer receives the modified embeddings and learns to represent the input text as a collection of capsules, where each capsule represents a particular characteristic or attribute of the text. The capsule layer refines capsule activations through multiple routing iterations by giving greater weight to capsules with higher agreement. The capsule outputs are then fed into dense layers in order to learn higher-level text representations. The final dense layer generates the output prediction, which indicates the classification or label of the input text. For training, the model is compiled with an appropriate optimizer and loss function. This architecture permits more expressive and hierarchical representations of the input text, thereby improving the model's ability to accurately classify text. Figure 1 shows the high-level diagram of the implemented CapsNet.

V. RESULTS

The accuracy acquired is 90.44%, which was higher than the CNN model. But it also couldn't predict two of the four test cases correctly which proved that CapsNet is not well-suited for text-related data. In CNN, neurons are activated to detect specific features, where they do not consider the properties of features like size, orientation, color, velocity, and so on. It is not trained on relationships between features and because MaxPooling in CNN, it loses information. MaxPooling is performed to achieve translation variance. A capsule in a CapsNet is made up of a layer of neurons that perform internal calculations to forecast the existence and the instantiation parameters of a specified feature at a particular spot.

DistilBERT is a lighter version of BERT [4] which uses fewer weights and achieves similar accuracy on several tasks with much lower training times (40% fewer parameters than bert-base-uncased, runs 60% faster). BERT (Bidirectional Encoder Representations from Transformers) is a paper published by researchers at Google, which caused a great stir in the NLP community as it became the state-of-the-art on several NLP tasks. BERT's key technical innovation was applying the bidirectional training of Transformer, a popular Attention model, to language modeling. This was in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training (such as LSTMs).

The paper's results showed that a language model which was bidirectionally trained can have a deeper sense of language context and flow than single-direction language models. In the paper, the researchers detailed a novel technique named Masked Language Model (MLM) which allows bidirectional training in models in which it was previously impossible. We used the BERT embeddings again, used them in our Transformer layer, which was then passed to a Dense layer with 500 neurons. We used a Dropout of 0.1 to avoid overfitting of data. We kept all the parameters the same as the other models previously described and achieved an accuracy of 93.24% (Table III), although it misclassified one of the test cases, which might be because we trained the model for 50 epochs, like the rest of the models. Figure 5 shows the workflow of all the models applied during this work.

VI. Conclusion

We investigated the dataset and characterized its sentiment, polarity, and readability. In addition, we have implemented a number of neural network models and demonstrated their performance using BERT embeddings for each of their backbones. Finally, a capsule neural network was designed for multiclass text classification, obtaining promising results that outperformed a CNN architecture with similar parameter counts. In addition, the DistilBERT model achieved the best results for this task, which inspired us to set a target and create a simple and effective model to address the current problem.

VII. Future Works

The work is still in progress, and we choose DistilBERT as the benchmark in order to improve CapsNet's text data performance over CNN. Instead of using ReLU, which is quite similar to ELU [3] but tends to converge cost to zero much more slowly, we will keep investigating using other hyperparameter adjustment techniques, which will likely yield more accurate finding during CapsNet training. In order to conduct a more thorough comparison and analysis and identify CapsNet's shortcomings, our research also involves adding Attention layers to the simulation and doing research with other models, such as LSTM and so on. As the dataset is multilingual and it is essential to solve the task from a multilingual perspective, we will work on a non-English or multilingual pre-trained model to improve the CapsNet architecture without relying on translation packages.

References

[1] Darya Bogoradnikova, Olesia Makhnytkina, Anton Matveev, Anastasia Zakharova, and Artem Akulov. Multilingual sentiment analysis and toxicity detection for text messages in russian. In *2021 29th Conference of Open Innovations Association (FRUCT)*, pages 55–64. IEEE, 2021.

[2] Theodora Chu, Kylie Jue, and Max Wang. Comment abuse classification with deep learning. *Von https://web. stanford. edu/class/cs224n/reports/2762092. pdf abgerufen*, 2016.

[3] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[4] 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.

[5] Kevin Khieu and Neha Narwal. Detecting and classifying toxic comments. *Web: https://web. stanford. edu/class/archive/cs/cs224n/cs224n*, 1184, 2017.

[6] Joao A Leite, Diego F Silva, Kalina Bontcheva, and Carolina Scarton. Toxic language detection in social media for brazilian portuguese: New dataset and multilingual analysis. *arXiv preprint arXiv:2010.04543*, 2020.

[7] Vittorio Mazzia, Francesco Salvetti, and Marcello Chiaberge. Efficient-capsnet: Capsule network with self-attention routing. *Scientific reports*, 11(1):14634, 2021.

[8] Huy Nguyen and Minh-Le Nguyen. A deep neural architecture for sentence-level sentiment classification in twitter social networking. In *Computational Linguistics: 15th International Conference of the Pacific Association for Computational Linguistics, PACLING 2017, Yangon, Myanmar, August 16–18, 2017, Revised Selected Papers 15*, pages 15–27. Springer, 2018.

[9] Guizhe Song, Degen Huang, and Zhifeng Xiao. A study of multilingual toxic text detection approaches under imbalanced sample distribution. *Information*, 12(5):205, 2021.

[10] Gaofei Xie. An ensemble multilingual model for toxic comment classification. In *International Conference on Algorithms, Microchips and Network Applications*, volume 12176, pages 429–433. SPIE, 2022.

[11] Jian Xu and Yuqing Zhai. A toxic comment classification model based on ensemble. In *Journal of Physics: Conference Series*, volume 1873, page 012080. IOP Publishing, 2021.

979-8-3503-0219-6/23/$31.00 ©2023 IEEE