Optimizations in Applied Machine Learning
Research Article* | *Volume 2* | *Issue 2* | *Aug 2022
Received: 2 July 2022 | Revised: 28 July 2022
Accepted: 8 Aug 2022 | Published Online: 10 Aug 2022

Improving the Applicability of Social Media Toxic Comments Prediction Across Diverse Data Platforms Using Residual Self-Attention-Based LSTM Combined with Transfer Learning

Jiahuai Ma[1], Zhaoyan Zhang[2*], Kaixian Xu[3], Yu Qiao[4]

[1] Department of Computer & Information Science, University of Florida, Florida, 32608, USA;

[2] Corresponding, Beijing Kwai Technology Co., Ltd, Beijing, 100085, China;

Email: zhaoyanzhang9394@gmail.com

[3] Risk & Quant Analytics, BlackRock, New Jersey, 10001, USA;

[4] Khoury College of Computer Sciences, Northeastern University, Washington, 98109, USA

Abstract: Toxic comments on social media, often involving hate speech and insults, pose significant challenges for online safety. While many studies have focused on detecting toxic comments within a single platform, cross-platform toxicity prediction remains underexplored. This task is particularly challenging due to linguistic differences and varying user behaviors across platforms, which reduce the effectiveness of models trained on one dataset when applied to another. To address these challenges, this paper proposes a Residual Self-Attention-Based LSTM framework with transfer learning. The model is first trained on a large source dataset (Twitter) and then fine-tuned on a smaller target dataset (YouTube). Residual connections ensure smooth gradient flow, while self-attention captures critical contextual features. Transfer learning enables the model to adapt to platform-specific nuances without retraining from scratch. Experiments show that the proposed approach significantly improves generalization across platforms, achieving higher precision, recall, and F1-scores compared to baseline methods. These results highlight the potential of combining advanced deep learning techniques with transfer learning for cross-platform toxicity detection, providing a foundation for future research in this area.

Keywords: *Social media toxic comments prediction; Transfer learning, machine learning; Residual connection.*

1. Introduction

Social media platforms have become indispensable tools for modern communication, connecting billions of users worldwide. From sharing personal experiences to engaging in global debates, these platforms enable unprecedented opportunities for interaction. However, alongside their positive impact, they have also facilitated the rise of harmful online behaviors, including the proliferation of toxic comments [1][2][3]. Toxic comments—often characterized by hate speech, cyberbullying, insults, or derogatory remarks—pose serious challenges to the well-being of individuals and the broader online community. Beyond their immediate impact on mental health, such comments can escalate into larger societal issues, fostering divisiveness and hostility. Recognizing the severity of this problem, researchers and practitioners have focused on devising

effective methods to identify and mitigate toxic content, making toxic comment prediction a vital task for creating safer digital spaces.

The process of detecting toxic comments is inherently complex. Unlike traditional methods [4][5][6], toxicity often resides in nuanced language that depends on context, tone, and cultural sensitivities. For example, a phrase that may seem benign in one cultural context can be deeply offensive in another. Similarly, sarcasm, slang, and emerging internet trends complicate the task of identifying harmful language. As a result, straightforward solutions like manual moderation or basic keyword filtering have proven insufficient in addressing the scale and subtlety of the problem. Manual moderation, while effective for small-scale platforms, is labor-intensive, prone to human error, and unsustainable for platforms with millions of active users. On the other hand, keyword-based approaches [7][8], which rely on predefined lists of offensive words, are rigid and frequently misclassify non-toxic comments as harmful or fail to detect cleverly disguised toxicity.

In response to these limitations, automated methods [9][10][11] for toxic comment detection have gained traction over the past decade. Early automated systems relied heavily on rule-based algorithms and handcrafted features, such as term frequency-inverse document frequency (TF-IDF) [12][13] and simple word embeddings. While these approaches improved efficiency compared to manual moderation, they still struggled with the contextual and dynamic nature of online language. The evolution of machine learning technologies [14][15][16], particularly in the field of natural language processing (NLP), has transformed toxic comment prediction by introducing models capable of learning intricate patterns from large datasets.

Machine learning models such as Support Vector Machines (SVM) [17][18], Random Forests [19][20], and Gradient Boosting [21][22] initially demonstrated notable improvements in toxic comment detection. However, their reliance on shallow feature representations limited their generalizability across diverse datasets. The advent of deep learning marked a paradigm shift in this domain, with models like convolutional neural networks (CNNs) [23][24] and recurrent neural networks (RNNs) [25][26] proving adept at handling complex tasks. For instance, Xiong et al. proposed an effective distributed data parallel acceleration-based generative adversarial network for fingerprint generation, demonstrating the superiority of the model [27]. More recently, transformer-based models like BERT and GPT [28][29][30] have set new benchmarks in NLP tasks, including toxic comment prediction, by leveraging attention mechanisms to capture long-range dependencies and subtle semantic nuances.

While these advancements are impressive, most research and applications have focused on achieving high accuracy on individual datasets or specific platforms. For example, a model trained on Twitter data often performs well within that domain but struggles to generalize when applied to comments from other platforms like YouTube, Reddit, or Facebook. This lack of cross-platform generalizability is a critical limitation, as toxic behaviors and linguistic styles vary significantly across platforms. A comment that might be flagged as toxic on Twitter due to its brevity and informal tone may go undetected on YouTube, where comments are often longer and follow different conversational norms. This gap in generalizability hinders the development of robust, universally applicable models, leaving platforms vulnerable to toxic content that falls outside their training scope.

The importance of addressing cross-platform generalizability cannot be overstated similar to the issues in other tasks [31][32]. Social media platforms often intersect, with users frequently sharing content across multiple channels. Toxic behaviors, too, are not confined to a single platform but

propagate across the digital ecosystem, exacerbating their societal impact. Developing models capable of adapting to diverse datasets is essential for creating holistic solutions that ensure safer interactions across the entire social media landscape. Furthermore, enhancing model generalizability is not just a technical challenge but also a step toward reducing biases in machine learning. Many current models inadvertently reflect the biases present in their training data, leading to unfair or inconsistent outcomes when applied to new datasets.

To address these challenges, this paper proposes a Residual Self-Attention-Based LSTM framework combined with transfer learning shown in Figure 1 to enhance toxic comment prediction across platforms. The approach begins with training on a source domain (e.g., Twitter) using a residual connection-enhanced LSTM module, which captures sequential dependencies while preserving critical information. A self-attention mechanism is applied to extract high-level contextual features, enabling the model to focus on relevant input aspects. To adapt the model to a target domain (e.g., YouTube), transfer learning is employed by fine-tuning only the fully connected layers while freezing pre-trained feature extraction layers. This strategy retains general knowledge of toxicity patterns while adapting to platform-specific nuances. The framework improves cross-platform applicability, reduces computational costs, and ensures efficient training.

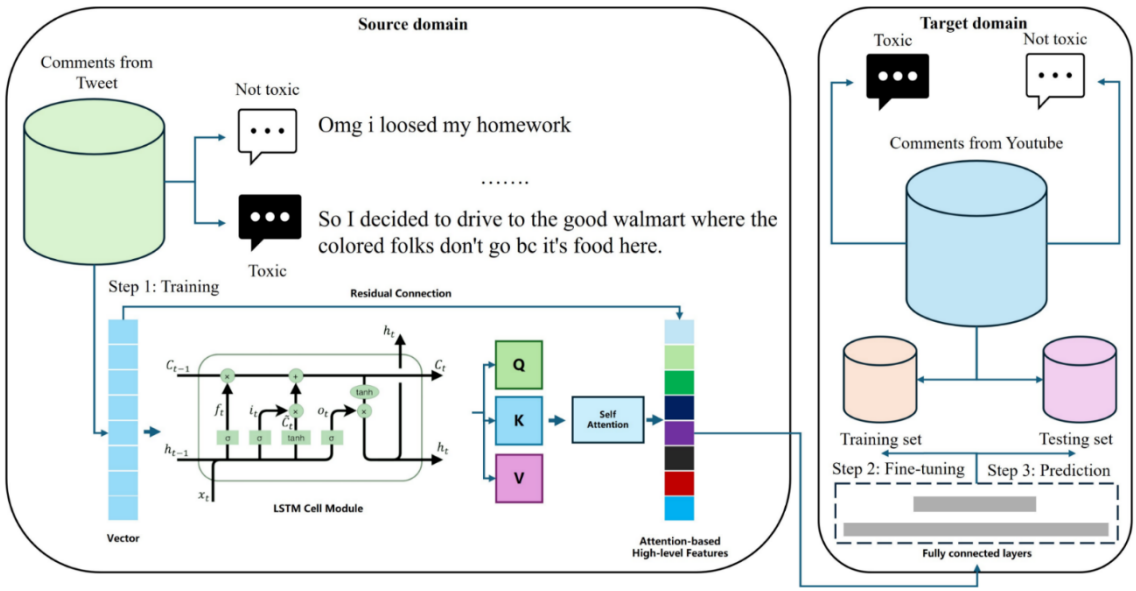

Figure 1. The workflow of the proposed model for social media toxic comments prediction across diverse data platforms.

2. Literature Review

A. Toxic comments prediction

The detection of toxicity in conversational data has been an active area of research, employing both traditional machine learning methods and advanced deep learning techniques due to their excellent performance in many tasks [33][34][35][36]. Traditional approaches often leverage classifiers such as Decision Trees [37], Logistic Regression [38], Support Vector Machines (SVM) [39], and Ensemble Models [40]. These models have proven effective in structured data scenarios

and are frequently applied beyond toxicity detection, such as in identifying other types of anti-social behaviors. For instance, Naive Bayes and Random Forests were successfully utilized to detect fake reviews on Amazon, relying on features like seller attributes, product details, and review content to classify fraudulent activity [41]. Similarly, [42] examined how the performance of classification models changes when applied in dynamic, real-world scenarios requiring online learning, revealing the challenges these methods face in maintaining robust performance under evolving data distributions.

In the domain of toxicity detection, more tailored approaches have been developed to handle the complexities of online communication. For example, [43] proposed a method for monitoring discussion threads to identify emerging aggressive behavior, using text representations alongside classifiers such as Radial Basis Function, SVMs, and Hidden Markov Models. These methods underscore the importance of capturing nuanced patterns in user interactions to predict potential toxicity accurately. Additionally, work by [44] focused on understanding how text length influences classification outcomes, particularly in detecting fake reviews, demonstrating that shorter texts often pose a greater challenge for model learning. Such findings emphasize the need for models capable of adapting to diverse text properties while maintaining high performance. In addition, deep learning models also attracted much attention due to its effectiveness in many tasks. In [45], the authors utilized Convolutional Neural Networks (CNNs) for multi-label classification of online comments, demonstrating the effectiveness of this approach by experimenting with various word embeddings to enhance feature representation and classification performance. Building on this, [46] conducted a comparative study evaluating the performance of CNNs against Long Short-Term Memory (LSTM) networks, highlighting the strengths and weaknesses of each architecture in capturing textual features for toxicity detection. Meanwhile, [47] introduced a novel approach leveraging Capsule Networks, which offer a hierarchical understanding of features, making them particularly useful for recognizing intricate relationships within toxic content. Monitoring the dynamics of toxicity in social networks adds another dimension to this research. For instance, [48] presented a CNN-based model designed to detect toxic tweets while incorporating temporal aspects. Their methodology extended beyond simple classification, utilizing hashtags associated with toxic tweets to analyze the propagation of toxicity over time. This approach not only identified toxic content but also provided insights into its spread and evolution within online communities, offering valuable perspectives for developing more proactive moderation systems.

3. Method

A. Dataset preparation

In this study, we utilized two datasets from Kaggle: one large dataset from Twitter as the source domain and another smaller dataset from YouTube, both focusing on toxic comment detection. The task at hand was a binary classification problem where the goal was to determine whether a given comment was toxic or not. The Twitter dataset comprised 56,745 samples, while the YouTube dataset contained 1,000 samples. The Twitter dataset was used for training dataset. The Youtube data was split into two parts: one half was used as the training set for fine-tuning the model through transfer learning, while the remaining portion served as the test set. Given that the data consisted of textual information, we applied text vectorization using Term Frequency-Inverse Document Frequency (TF-IDF). This method transforms raw text into numerical vectors, making it suitable for input into machine learning models. We restricted the number of features to 50 using the vectorizer to reduce the dimensionality of the data while retaining significant information.

Figure 2 and Figure 3 illustrate the label distribution of both datasets. Since the overall distribution was almost balanced, no further balancing techniques were applied.

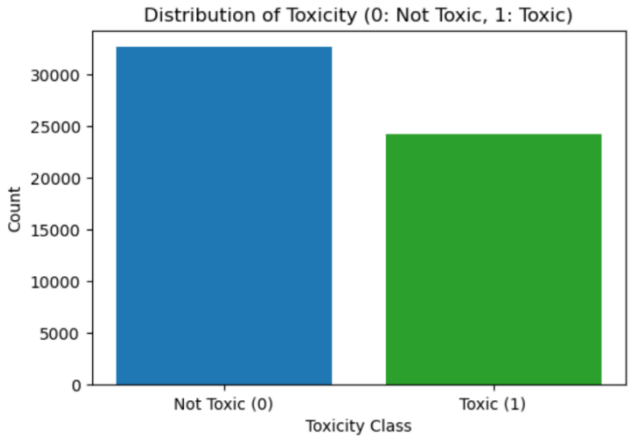

Figure 2. The label data distribution of the Twitter dataset.

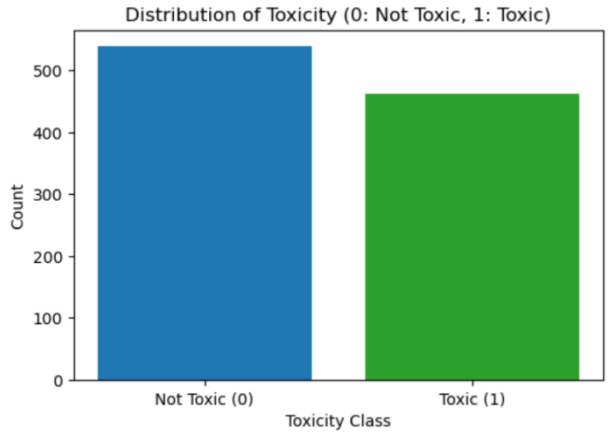

Figure 3. The label data distribution of the YouTube dataset.

B. The residual self-attention-based LSTM model

1. Preliminaries of the LSTM

Long Short-Term Memory (LSTM) networks are a specialized type of recurrent neural network (RNN) designed to handle complex sequence prediction problems by learning long-term dependencies within data. Unlike conventional feedforward neural networks, LSTMs possess feedback connections that enable them to process entire sequences effectively. This unique characteristic makes LSTMs particularly well-suited for tasks involving sequential data, such as time series forecasting, natural language processing, and speech recognition.

The concept of LSTM was introduced by Hochreiter and Schmidhuber in 1997 to tackle a fundamental issue faced by traditional RNNs: the vanishing gradient problem. This problem arises during backpropagation when gradients, which are propagated backward through the network across multiple layers, diminish exponentially over time. This shrinkage makes it difficult for the network to learn and model long-range dependencies or correlations between distant events. LSTMs address this challenge by incorporating specialized memory cells that can maintain and

manage information for extended periods, ensuring that significant data is retained and utilized effectively. An LSTM unit consists of three types of gates: the input gate, the forget gate, and the output gate. These gates govern the flow of information through the network, determining which data is updated, retained, or discarded. More information is provided as follows: 1) Input Gate: This gate controls the degree to which new information is added to the cell's memory. It uses a sigmoid activation function to decide which values should be allowed to modify the memory state, paired with a tanh function that generates a set of candidate values for updating the memory. 2) Forget Gate: This gate is responsible for deciding which information from the cell's memory should be removed. Based on the current input and the previous output, the forget gate enables the model to discard outdated or irrelevant data, preventing information overload and promoting efficient learning. 3) Output Gate: The output gate determines how much of the information stored in the cell's memory should contribute to the final output of the LSTM unit. It evaluates the current input and the preceding output to decide which portion of the cell state is passed forward and used in the subsequent computation.

2. Preliminaries of the self-attention

Self-attention, a pivotal element of the transformer architecture, is an innovative mechanism that enables a model to independently assess the importance of different parts of an input sequence, regardless of their position. This has dramatically transformed fields such as natural language processing (NLP) and other tasks that involve sequential data by providing a versatile and efficient way to handle complex input information. The essence of self-attention lies in evaluating the relevance of all components of the input to each specific output element. In practical terms, this means that each output unit—such as a word in a sentence—can be represented as a weighted combination of all input units, with the weights indicating the importance of each input to the computation of the output.

The self-attention mechanism operates through three distinct vectors for each input element: the Query, the Key, and the Value. These vectors are created by applying learned linear transformations to the input elements. To compute attention for a given element, its Query vector is matched against the Key vectors of all other elements in the sequence to generate scores. These scores, which determine the importance of the corresponding Value vectors, are then normalized using a softmax function to create a distribution of attention weights that sum to one.

This mechanism allows the model to selectively focus on the most relevant sections of the input sequence, adapting its attention based on context. Such adaptability is especially beneficial for tasks like machine translation, where the relationships between input words can shift considerably based on the context. By dynamically computing attention for each input-output pair, self-attention can emphasize or minimize features as necessary, unlike recurrent layers that operate within a fixed order.

Additionally, self-attention supports parallel processing, which accelerates training compared to the sequential nature of RNNs. This parallelization, along with its capacity to handle long-range dependencies within data, establishes self-attention as an essential component in modern neural network architectures, paving the way for powerful and efficient models that achieve state-of-the-art performance across a range of tasks.

3. Preliminaries of the residual connection

Residual connections, also known as skip connections, are an architectural innovation in deep learning that helps overcome the challenges associated with training very deep neural networks. One of the main difficulties of training deep networks is the vanishing gradient problem, where gradients become so small during backpropagation that they fail to make meaningful updates to the network's parameters, leading to stagnant training and poor convergence. Residual connections mitigate this issue by creating shortcuts that allow gradients to flow more effectively through the network, bypassing one or more layers.

These connections were popularized by He et al. in their groundbreaking work on ResNet, which demonstrated that deep networks could be trained effectively by using residual structures. The core concept behind residual connections is straightforward: instead of directly learning a transformation from input to output, a layer learns the difference between the input and the output, known as the residual. Mathematically, this is expressed as $F(x)+x$ where x is the input to the layer, and $F(x)$ is the output of the transformation applied by the layer. The result of this operation is added back to the original input, creating a direct path for the signal. This setup forms a shortcut path for the backward pass during training, ensuring that the gradient can be passed directly through the network without diminishing significantly. This makes it possible to train much deeper networks than was previously possible, as each layer only needs to learn the adjustments required rather than the full transformation. Consequently, residual connections simplify the learning process, improve the flow of gradients, and lead to more efficient training of deep neural networks.

4. The architecture of the proposed model

The architecture of the proposed model starts with an input layer that receives a sequence of data, which is typically vectorized text input. This input passes through an embedding layer with output dimension of 16, which transforms the raw integer-encoded input into dense vector representations, capturing semantic relationships between words. The output of the embedding layer is then processed by a bidirectional LSTM layer with 16 neurons and relu activation function, which enhances the model's ability to learn sequential dependencies by processing the input in both forward and backward directions. The bidirectional LSTM output is further refined by a self-attention layer, which allows the model to weigh different parts of the input sequence differently based on their relevance. This attention mechanism helps the model focus on important sections of the input for better performance in tasks like sequence classification. The output of the self-attention mechanism is combined with the original output from the bidirectional LSTM through a residual connection, aiding in smoother gradient flow and more efficient training by bypassing some layers. Finally, the refined features are passed through a series of fully connected dense layers with 8 and 1 neurons, with the last layer using a sigmoid activation function to output a binary classification, indicating whether a given input is toxic or non-toxic.

5. Transfer learning-based prediction

Transfer learning is a powerful technique in machine learning where knowledge gained from training a model on one task (the source domain) is applied to a different but related task (the target domain). This approach is particularly beneficial when the target task has limited data available, as it allows the model to leverage existing knowledge from the source task, reducing the need for extensive training from scratch. By reusing learned features and representations, transfer learning helps improve the efficiency and effectiveness of training models for new tasks.

After the initial training phase, transfer learning is used to modify the model for the target domain (e.g., YouTube). This process involves fine-tuning only the fully connected layers while keeping

the pre-trained feature extraction layers frozen. By doing so, the model retains the valuable knowledge and learned representations from the source domain while dedicating its capacity to adapt its predictions to the unique features of the target domain. In addition, this study uses TensorFlow for model training, employing the Adam optimizer for efficient optimization. The loss function utilized is binary cross-entropy, which is well-suited for binary classification tasks such as detecting toxic comments. To evaluate model performance, metrics including precision, recall, F1-score, and accuracy are used. These evaluation metrics provide a comprehensive understanding of the model's ability to correctly identify toxic comments while balancing false positives and false negatives.

4. Results and Discussion

A. The performance of the model

Figure 4 shows the training and validation performance of our proposed model. The training accuracy and loss curves are based on the Tweet dataset, which was used as the source domain for initial training. The validation curves, on the other hand, were generated using 50% training dataset of the previously split YouTube dataset, which was designated as the validation set to monitor the model's performance and save the best weights. In the accuracy plot on the left, the training accuracy (green line) steadily increases over the epochs, showing that the model is effectively learning patterns from the Tweet dataset. However, the validation accuracy (dashed orange line) fluctuates throughout the training process, likely due to the smaller size and different characteristics of the YouTube dataset. This indicates that the model struggles to fully adapt to the target domain, which may be an early sign of overfitting, where the model starts to perform better on the training data but not as well on new data. In the loss plot on the right, the training loss (blue line) decreases consistently, reflecting that the model is minimizing the objective function on the Tweet data. However, the validation loss (dashed red line) remains relatively stable but shows less improvement compared to the training loss, which further suggests potential overfitting.

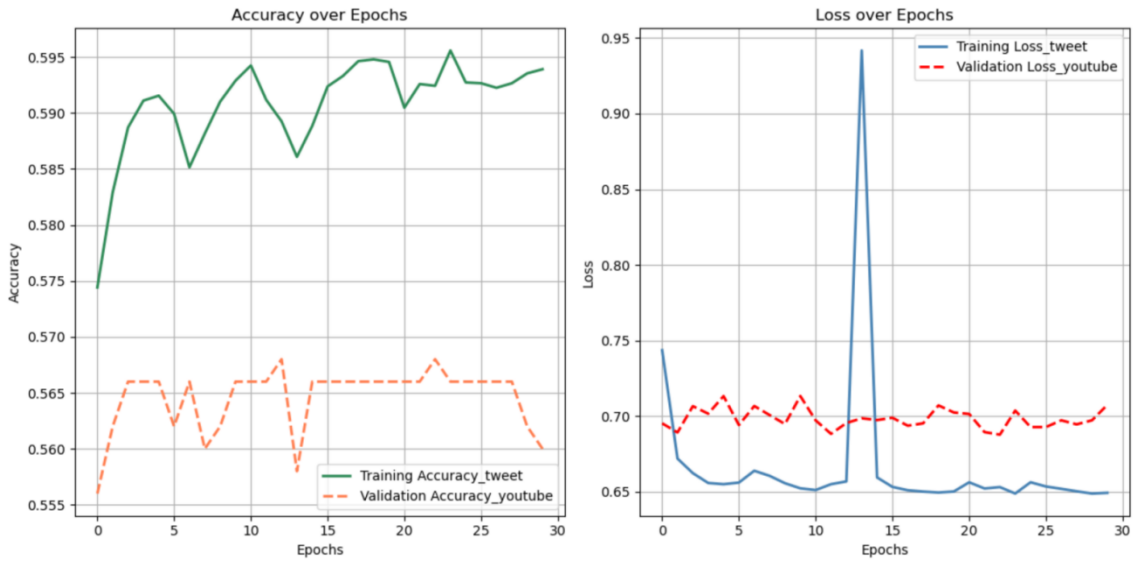

Figure 4. The training curve of the proposed model without transfer learning.

The experimental results shown in Table 1, Figure 5 and Figure 6 demonstrate the performance differences between the Residual Self-Attention-Based LSTM and the Residual Self-Attention-Based LSTM+Transfer Learning models when applied to the YouTube dataset for toxicity detection. The Residual Self-Attention-Based LSTM model, trained solely on the Twitter dataset, exhibited limited generalization to the YouTube dataset. This is evident in its performance metrics, where the recall was particularly low at 0.0082, indicating its inability to capture most toxic samples in the target domain. The precision was 0.5000, suggesting a high rate of false positives, while the F1-score, which balances precision and recall, was only 0.0162. The overall accuracy of this model was 0.5140, demonstrating its poor predictive capability when applied to a dataset with different domain characteristics. The confusion matrix further reveals the significant number of toxic samples that were misclassified as non-toxic, emphasizing the model's difficulty in adapting to the YouTube data.

In contrast, the Residual Self-Attention-Based LSTM+Transfer Learning model, which was fine-tuned on a portion of the YouTube training set after initial training on the Twitter dataset, showed substantial improvements across all evaluation metrics. The recall increased to 0.2551, reflecting the model's enhanced ability to identify toxic comments. Additionally, the precision rose to 0.9688, indicating that the majority of toxic predictions were accurate, thereby reducing the number of false positives. This improvement in both recall and precision led to a significant increase in the F1-score to 0.4039, highlighting the model's balanced performance in predicting toxic and non-toxic samples. The accuracy also improved to 0.6340, confirming the model's better overall effectiveness in the target domain. The confusion matrix for this model shows a notable reduction in false negatives and false positives, with more toxic comments correctly classified as such.

Table 1. The performance of the model with and without transfer learning in YouTube testing dataset prediction

Model Name	Precision	Recall	Accuracy	F1-score
Residual Self-Attention-Based LSTM	0.5000	0.0082	0.5140	0.0162
Residual Self-Attention-Based LSTM+transfer learning	0.9688	0.2551	0.6340	0.4039

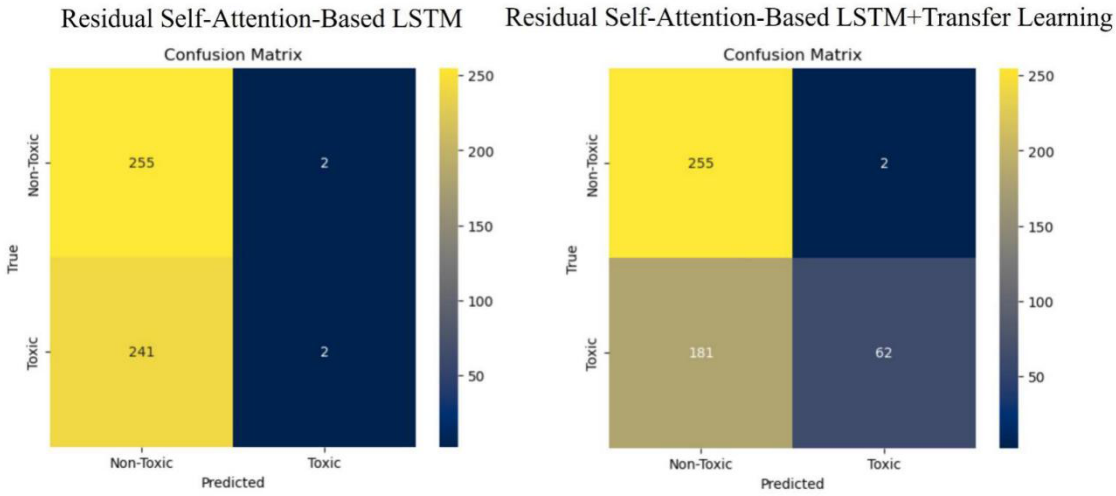

Figure 5. The confusion matrix of the model with and without transfer learning in YouTube testing dataset prediction

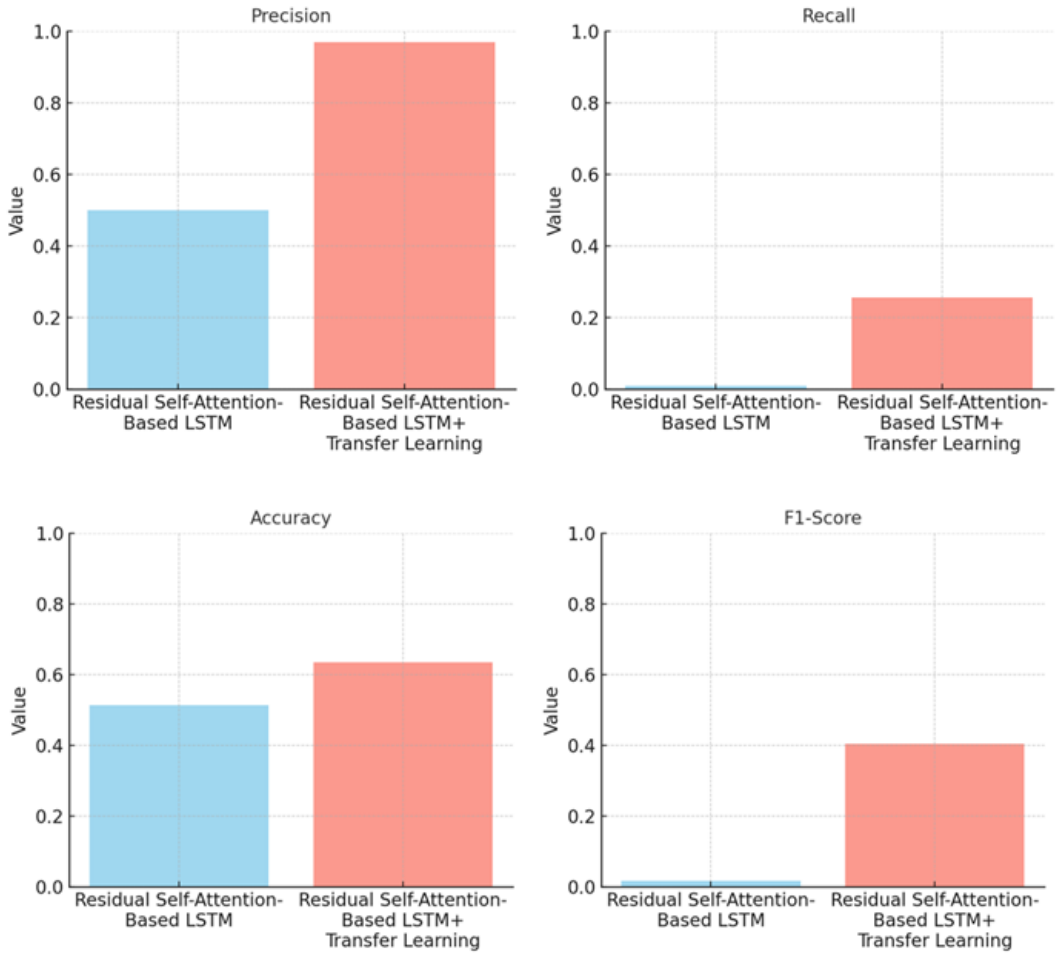

Figure 6. The visualization of performance of the model with and without transfer learning in YouTube testing dataset prediction

Figure 7 represents the outputs of the first fully connected layer from the models, reduced to two dimensions using PCA for better interpretability. On the left, the Residual Self-Attention-Based LSTM model, trained exclusively on the Twitter dataset and applied directly to the YouTube dataset, shows a relatively compact clustering of feature representations. However, the data points are not well-separated, indicating limited ability to discriminate between toxic and non-toxic samples. This reflects the model's struggle to generalize across domains, as seen in the earlier metrics. On the right, the Residual Self-Attention-Based LSTM+Transfer Learning model demonstrates a more diverse and better-distributed feature space. This model, which underwent fine-tuning on the YouTube dataset after initial training on the Twitter dataset, clearly benefits from transfer learning. The representation shows improved separability and richer clustering, which indicates that the model has adapted its feature extraction process to better capture the nuances of the YouTube dataset. This adaptation enables the model to distinguish between toxic and non-toxic samples more effectively, resulting in improved performance metrics.

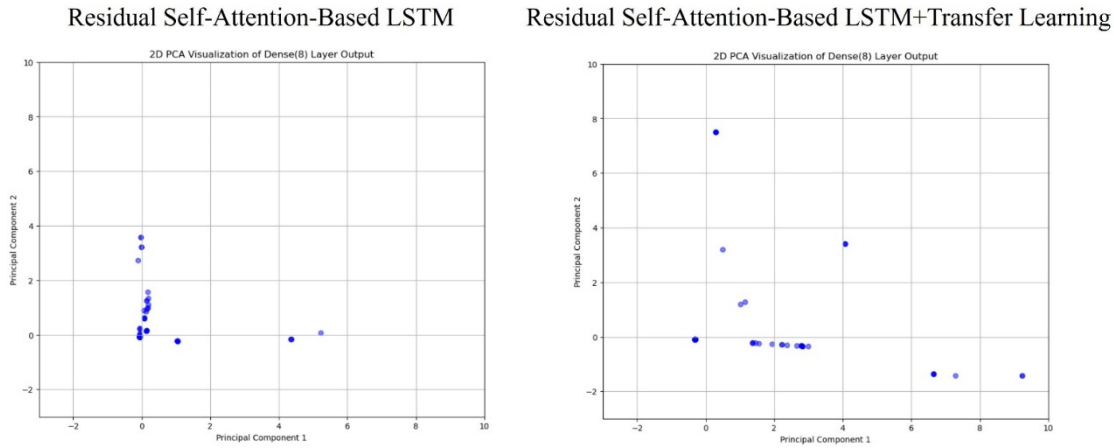

Figure 7. The visualization of the learned high-level representations from the first fully connected layer of the model.

B. The influence of different components on the model performance

The results presented in Table 2 and the accompanying visualizations shown in Figure 8 and Figure 9 illustrate the impact of different components on the performance of the models for toxicity detection. Three models were evaluated: LSTM+Transfer Learning, Self-Attention-Based LSTM+Transfer Learning, and Residual Self-Attention-Based LSTM+Transfer Learning.

The LSTM+Transfer Learning model serves as the baseline and leverages the sequential modeling capabilities of LSTM networks. While it achieves a high precision of 0.9608, its recall is notably low at 0.2016, indicating its limited ability to identify toxic samples. The accuracy of 0.6080 reflects reasonable performance on the majority class, but the F1-score of 0.3333 highlights an imbalance between precision and recall. The confusion matrix reveals a significant number of false negatives, suggesting that the sequential modeling alone struggles to capture nuanced relationships in the target domain without additional architectural enhancements.

The introduction of a self-attention mechanism in the Self-Attention-Based LSTM+Transfer Learning model significantly improves the model's performance. Self-attention enables the network to assign varying levels of importance to different parts of the input sequence, allowing it

to capture complex dependencies and contextual information more effectively. This model achieves a recall of 0.2346, an improvement over the baseline, indicating enhanced detection of toxic samples. The precision increases slightly to 0.9661, and the accuracy rises to 0.6240, reflecting better overall generalization. The F1-score improves to 0.3775, highlighting a more balanced trade-off between precision and recall. The confusion matrix demonstrates a reduction in false negatives compared to the baseline, validating the contribution of the self-attention mechanism in refining the model's understanding of input sequences. The Residual Self-Attention-Based LSTM+Transfer Learning model further enhances performance by incorporating residual connections alongside the self-attention mechanism. Residual connections address the issue of gradient vanishing and allow the model to retain critical information across layers, ensuring better feature propagation.

Table 2. The influence of different components on the model performance

Model Name	Precision	Recall	Accuracy	F1-score
LSTM+transfer learning	0.9608	0.2016	0.6080	0.3333
Self-Attention-Based LSTM+transfer learning	0.9661	0.2346	0.6240	0.3775
Residual Self-Attention-Based LSTM+transfer learning	0.9688	0.2551	0.6340	0.4039

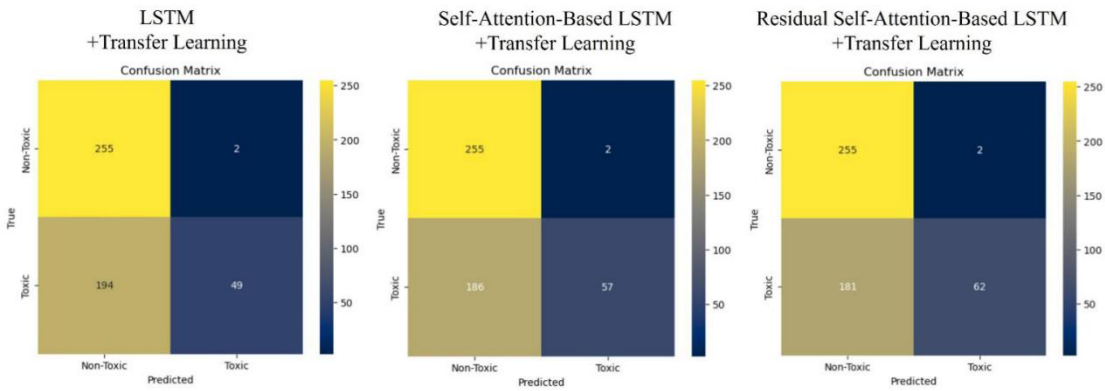

Figure 8. The confusion matrix related to the influence of different components on the model performance

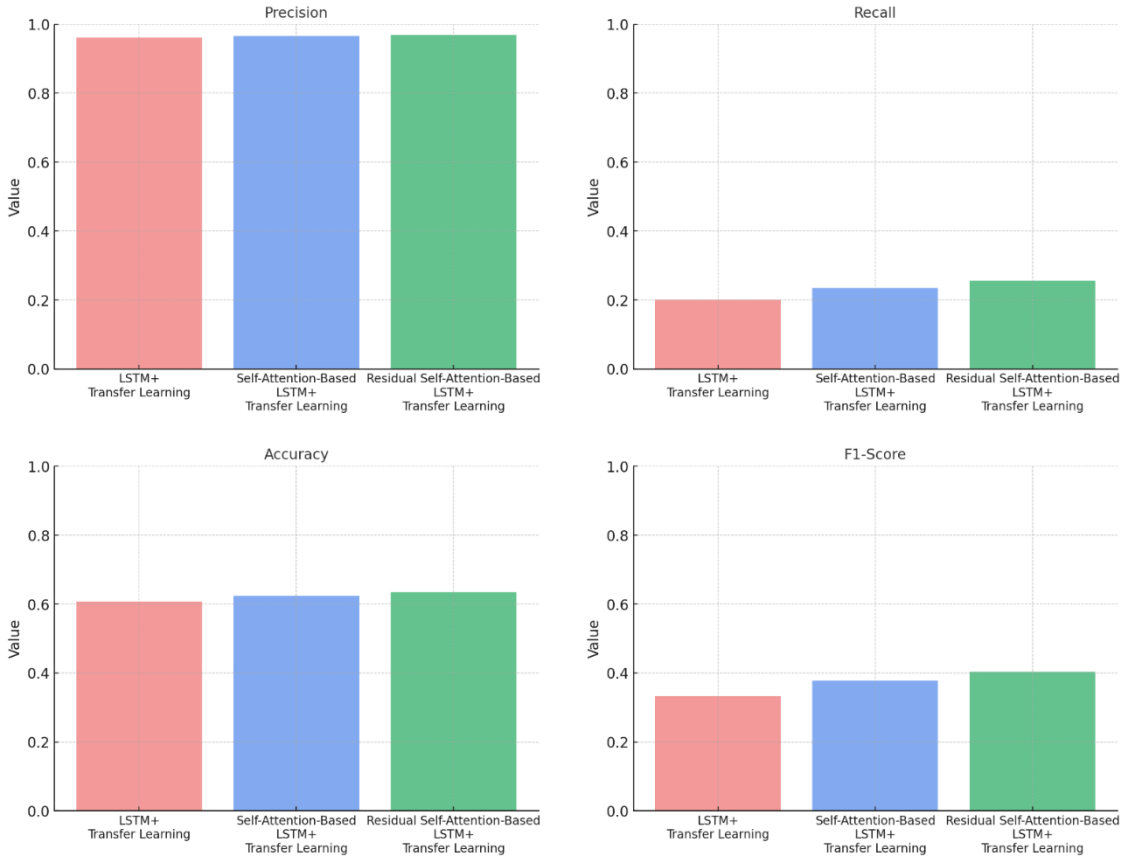

Figure 9. The visualization of the influence of different components on the model performance

C. Discussion

The Residual Self-Attention-Based LSTM+Transfer Learning model demonstrates notable improvements in cross-domain toxicity detection, particularly when compared to its simpler counterparts. However, despite its superior performance, the model still faces limitations that require attention in future research. One primary limitation lies in the reliance on a large amount of labeled data in the source domain (Twitter) to pre-train the model, as well as a subset of labeled data from the target domain (YouTube) for fine-tuning. This dependency makes the approach less feasible for applications where labeled data is scarce or expensive to obtain. Additionally, while the inclusion of self-attention and residual connections enhances the model's ability to capture contextual and long-range dependencies, it also increases the complexity and computational overhead, which may not be suitable for resource-constrained environments or real-time applications. In the future, the advanced methods from other domains should be considered for further improvement of the model performance [49][50].

5. Conclusion

This study highlights the challenges of cross-platform toxicity detection, particularly the variability in linguistic patterns and user behaviors across different social media platforms. The proposed Residual Self-Attention-Based LSTM+Transfer Learning model effectively addresses these issues by leveraging self-attention for contextual understanding and transfer learning for domain

adaptation. By fine-tuning on a small target dataset, the model retains general knowledge while adapting to platform-specific characteristics, resulting in improved performance across all key metrics. However, challenges remain, such as the reliance on labeled data for both source and target domains and the computational cost of advanced architectures. Future work should explore unsupervised transfer learning and lightweight model designs to enhance scalability and reduce dependency on labeled data. Despite these limitations, the study demonstrates the importance of cross-platform research and offers a promising approach for developing adaptable and effective toxicity detection models.

Funding
Not applicable

Author Contributions
Jiahuai Ma contributed to the conceptualization and design of the study, as well as the analysis and interpretation of data. Zhaoyan Zhang (*corresponding author*) led the writing of the manuscript and oversaw the overall development of the research project. Kaixian Xu assisted with data collection and methodology, contributing to the writing and critical revision of the manuscript. Yu Qiao supported the data analysis and contributed to manuscript revisions and editing.

Institutional Reviewer Board Statement
Not applicable

Informed Consent Statement
Not applicable

Data Availability Statement
The data supporting the findings of this study are available from the corresponding author upon request.

Conflict of Interest
The authors declare no conflict of interest.

References

[1] Zaheri, S., Leath, J., & Stroud, D. (2020). Toxic comment classification. SMU Data Science Review, 3(1), 13.

[2] Van Aken, B., Risch, J., Krestel, R., & Löser, A. (2018). Challenges for toxic comment classification: An in-depth error analysis. arXiv preprint arXiv:1809.07572.

[3] Risch, J., & Krestel, R. (2020). Toxic comment detection in online discussions. Deep learning-based approaches for sentiment analysis, 85-109.

[4] Zhu, D., Gan, Y., & Chen, X. (2021). Domain Adaptation-Based Machine Learning Framework for Customer Churn Prediction Across Varing Distributions. Journal of Computational Methods in Engineering Applications, 1-14.

[5] Zhou, Z., Wu, J., Cao, Z., She, Z., Ma, J., & Zu, X. (2021, September). On-Demand Trajectory Prediction Based on Adaptive Interaction Car Following Model with Decreasing Tolerance. In 2021 International Conference on Computers and Automation (CompAuto) (pp. 67-72). IEEE.

[6] Wang, H., Li, J., & Xiong, S. (2008). Efficient join algorithms for distributed information integration based on XML. International Journal of Business Process Integration and Management, 3(4), 271-281.

[7] Brassard-Gourdeau, E., & Khoury, R. (2019, August). Subversive toxicity detection using sentiment information. In Proceedings of the Third Workshop on Abusive Language Online (pp. 1-10).

[8] Yee, K., Sebag, A. S., Redfield, O., Sheng, E., Eck, M., & Belli, L. (2022). A keyword based approach to understanding the over penalization of marginalized groups by English marginal abuse models on Twitter. arXiv preprint arXiv:2210.06351.

[9] Xiong, S., & Li, J. (2009, April). Optimizing many-to-many data aggregation in wireless sensor networks. In Asia-Pacific Web Conference (pp. 550-555). Berlin, Heidelberg: Springer Berlin Heidelberg.

[10] Xiong, S., & Li, J. (2010, June). An efficient algorithm for cut vertex detection in wireless sensor networks. In 2010 IEEE 30th International Conference on Distributed Computing Systems (pp. 368-377). IEEE.

[11] Li, J., & Xiong, S. (2010). Efficient Pr-skyline query processing and optimization in wireless sensor networks. Wireless Sensor Network, 2(11), 838.

[12] Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures. Information Processing & Management, 39(1), 45-65.

[13] Qaiser, S., & Ali, R. (2018). Text mining: use of TF-IDF to examine the relevance of words to documents. International Journal of Computer Applications, 181(1), 25-29.

[14] Yu, L., Li, J., Cheng, S., & Xiong, S. (2011, April). Secure continuous aggregation via sampling-based verification in wireless sensor networks. In 2011 Proceedings IEEE INFOCOM (pp. 1763-1771). IEEE.

[15] Xiong, S., Li, J., Li, M., Wang, J., & Liu, Y. (2011, April). Multiple task scheduling for low-duty-cycled wireless sensor networks. In 2011 Proceedings IEEE INFOCOM (pp. 1323-1331). IEEE.

[16] Feng, Z., Xiong, S., Cao, D., Deng, X., Wang, X., Yang, Y., ... & Wu, G. (2015, March). Hrs: A hybrid framework for malware detection. In Proceedings of the 2015 ACM International Workshop on International Workshop on Security and Privacy Analytics (pp. 19-26).

[17] Jakkula, V. (2006). Tutorial on support vector machine (svm). School of EECS, Washington State University, 37(2.5), 3.

[18] Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., & Xu, W. (2018). Applications of support vector machine (SVM) learning in cancer genomics. Cancer genomics & proteomics, 15(1), 41-51.

[19] Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

[20] Biau, G. (2012). Analysis of a random forests model. The Journal of Machine Learning Research, 13(1), 1063-1095.

[21] Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. Frontiers in neurorobotics, 7, 21.

[22] Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54, 1937-1967.

[23] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems, 33(12), 6999-7019.

[24] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. Pattern recognition, 77, 354-377.

[25] Medsker, L. R., & Jain, L. (2001). Recurrent neural networks. Design and Applications, 5(64-67), 2.

[26] Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2017). Recent advances in recurrent neural networks. arXiv preprint arXiv:1801.01078.

[27] Xiong, S., Zhang, H., Wang, M., & Zhou, N. (2022). Distributed Data Parallel Acceleration-Based Generative Adversarial Network for Fingerprint Generation. Innovations in Applied Engineering and Technology, 1-12.

[28] Ghojogh, B., & Ghodsi, A. (2020). Attention mechanism, transformers, BERT, and GPT: tutorial and survey.

[29] Qu, Y., Liu, P., Song, W., Liu, L., & Cheng, M. (2020, July). A text generation and prediction system: pre-training on new corpora using BERT and GPT-2. In 2020 IEEE 10th international conference on electronics information and emergency communication (ICEIEC) (pp. 323-326). IEEE.

[30] Topal, M. O., Bas, A., & van Heerden, I. (2021). Exploring transformers in natural language generation: Gpt, bert, and xlnet. arXiv preprint arXiv:2102.08036.

[31] Dai, W. (2022). Evaluation and improvement of carrying capacity of a traffic system. Innovations in Applied Engineering and Technology, 1-9.

[32] Zhao, Z., Ren, P., & Tang, M. (2022). Analyzing the Impact of Anti-Globalization on the Evolution of Higher Education Internationalization in China. Journal of Linguistics and Education Research, 5(2), 15-31.

[33] Lei, J. (2022). Green Supply Chain Management Optimization Based on Chemical Industrial Clusters. Innovations in Applied Engineering and Technology, 1-17.

[34] Lei, J. (2022). Efficient Strategies on Supply Chain Network Optimization for Industrial Carbon Emission Reduction. Journal of Computational Methods in Engineering Applications, 1-11.

[35] Dai, W. (2021). Safety evaluation of traffic system with historical data based on Markov process and deep-reinforcement learning. Journal of Computational Methods in Engineering Applications, 1-14.

[36] Xiong, S., Zhang, H., & Wang, M. (2022). Ensemble Model of Attention Mechanism-Based DCGAN and Autoencoder for Noised OCR Classification. Journal of Electronic & Information Systems, 4(1), 33-41.

[37] Shtovba, S., Shtovba, O., & Petrychko, M. (2019). Detection of social network toxic comments with usage of syntactic dependencies in the sentences. In CEUR Workshop Proceedings (pp. 313-323).

[38] Saif, M. A., Medvedev, A. N., Medvedev, M. A., & Atanasova, T. (2018, December). Classification of online toxic comments using the logistic regression and neural networks models. In AIP conference proceedings (Vol. 2048, No. 1). AIP Publishing.

[39] Rupapara, V., Rustam, F., Shahzad, H. F., Mehmood, A., Ashraf, I., & Choi, G. S. (2021). Impact of SMOTE on imbalanced text features for toxic comments classification using RVVC model. IEEE Access, 9, 78621-78634.

[40] Haralabopoulos, G., Anagnostopoulos, I., & McAuley, D. (2020). Ensemble deep learning for multilabel binary classification of user-generated content. Algorithms, 13(4), 83.

[41] Chowdhary, N. S., & Pandit, A. A. (2018). Fake review detection using classification. Int. J. Comput. Appl, 180(50), 16-21.

[42] Cardoso, E. F., Silva, R. M., & Almeida, T. A. (2018). Towards automatic filtering of fake reviews. Neurocomputing, 309, 106-116.

[43] Ventirozos, F. K., Varlamis, I., & Tsatsaronis, G. (2018). Detecting aggressive behavior in discussion threads using text mining. In Computational Linguistics and Intelligent Text Processing: 18th International Conference, CICLing 2017, Budapest, Hungary, April 17–23, 2017, Revised Selected Papers, Part II 18 (pp. 420-431). Springer International Publishing.

[44] Machová, K., Mach, M., & Demková, G. (2020). Modelling of the fake posting recognition in on-line media using machine learning. In SOFSEM 2020: Theory and Practice of Computer Science: 46th International Conference on Current Trends in Theory and Practice of Informatics, SOFSEM 2020, Limassol, Cyprus, January 20–24, 2020, Proceedings 46 (pp. 667-675). Springer International Publishing.

[45] Mestry, S., Singh, H., Chauhan, R., Bisht, V., & Tiwari, K. (2019, April). Automation in social networking comments with the help of robust fasttext and cnn. In 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT) (pp. 1-4). IEEE.

[46] Anand, M., & Eswari, R. (2019, March). Classification of abusive comments in social media using deep learning. In 2019 3rd international conference on computing methodologies and communication (ICCMC) (pp. 974-977). IEEE.

[47] Srivastava, S., Khurana, P., & Tewari, V. (2018, August). Identifying aggression and toxicity in comments using capsule network. In Proceedings of the first workshop on trolling, aggression and cyberbullying (TRAC-2018) (pp. 98-105).

[48] Georgakopoulos, S. V., Tasoulis, S. K., Vrahatis, A. G., & Plagianakos, V. P. (2018, July). Convolutional neural networks for toxic comment classification. In Proceedings of the 10th hellenic conference on artificial intelligence (pp. 1-6).

[49] Yu, L., Li, J., Cheng, S., Xiong, S., & Shen, H. (2013). Secure continuous aggregation in wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems, 25(3), 762-774.

[50] Xiong, S., Yu, L., Shen, H., Wang, C., & Lu, W. (2012, March). Efficient algorithms for sensor deployment and routing in sensor networks for network-structured environment monitoring. In 2012 Proceedings IEEE INFOCOM (pp. 1008-1016). IEEE.

© The Author(s) 2022. Published by Hong Kong Multidisciplinary Research Institute (HKMRI).

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.