Cyber-Troll Detection using Deep Learning and NLP : A Comparative Study

Djibrim Mahaman Tahir M. Atto[1] ; Jaouhar Fattahi[2] ; Mohamed Mejri[2] ; Brahim Hnich[1] and Abdoul Majid O. Thiombiano[1]

***Abstract*— The proliferation of malicious online behaviors, particularly cyber-trolling, presents significant challenges to maintaining healthy online communities. This paper investigates the efficacy of four deep learning architectures—BERT, LSTM, GRU, and Causal Convolutional Networks (Causal Conv 1D)—for the automatic detection of cyber-trolls based on textual content. Using a comprehensive dataset of 50,000 social media comments, we evaluate these models on their ability to distinguish between normal users and trolls. Our results indicate that while the pre-trained BERT model achieves the highest overall accuracy (94.2%), the Causal Conv 1D architecture demonstrates competitive performance (92.7%) with significantly lower computational requirements. We also analyze the semantic features that most effectively contribute to troll detection and discuss the ethical implications of automated moderation systems. This research contributes to the development of more efficient and effective methods for maintaining civil discourse in online spaces.**

***Index Terms*— cyber-trolls, deep learning, natural language processing, BERT, LSTM, GRU, causal convolution, social media analysis**

I. Introduction

Detecting criminal activity online is not a new concept but how it can occur is changing. Technology and the influx of social media applications and platforms has a vital part to play in this changing landscape. As such, we observe an increasing problem with cyber abuse and 'trolling'/toxicity amongst social media platforms sharing stories, posts, memes sharing content [1]–[9]. Traditionally, detection of cyber-trolls has relied on manual content moderation, which faces significant scalability challenges given the volume of online content. Rule-based automated systems, while faster, often fail to capture the contextual nuances and evolving patterns of troll behavior [10]. Recent advances in deep learning and natural language processing (NLP) offer promising solutions to these challenges by enabling more sophisticated analysis of linguistic patterns and behavioral cues associated with trolling [11].

In this study, we evaluate four deep learning architectures for cyber-troll detection:

[1]Djibrim Mahaman Tahir M. Atto and Brahim Hnich and Abdoul Majid O. Thiombiano are with the Faculty of Science, University of Monastir, Tunisia. attodjibrim.mahamantahirm@fsegma.u-monastir.tn; brahim.hnich@fsm.rnu.tn; abdoulmajid.ousseinithiombiano@fsm.rnu.tn

[2]Jaouhar Fattahi and Mohamed Mejri are with the Department of Computer Science and Software Engineering, Laval University, Quebec, Canada. Jaouhar.Fattahi.1@ulaval.ca; Mohamed.Mejri@ift.ulaval.ca

- **BERT (Bidirectional Encoder Representations from Transformers)**: A transformer-based model that processes text bidirectionally, capturing contextual relationships in both directions [12].

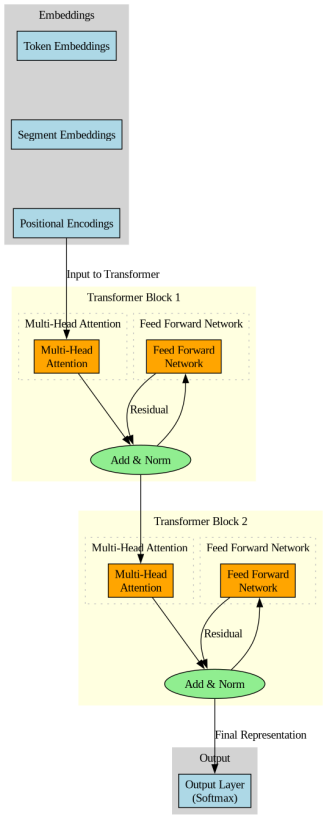


Fig. 1. BERT architecture

- **LSTM (Long Short-Term Memory)**: A recurrent neural network architecture designed to model sequential data with long-term dependencies [13].
The LSTM cell updates its states using the following equations:

$$\begin{aligned} f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ \tilde{C}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ h_t &= o_t \odot \tanh(C_t) \end{aligned} \tag{1}$$

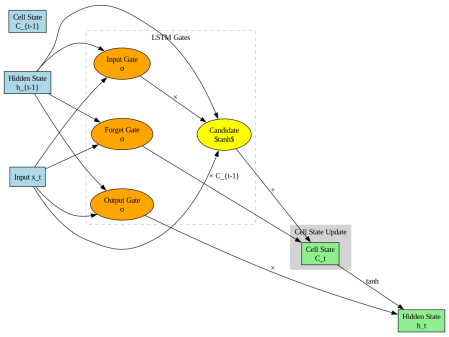


Fig. 2. LSTM architecture

- **GRU (Gated Recurrent Unit)**: A streamlined variant of LSTM that uses fewer parameters while maintaining similar capabilities [14].

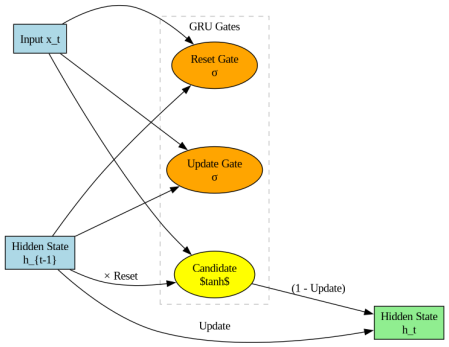


Fig. 3. GRU architecture

The GRU cell is governed by the following equations:

$$
\begin{aligned}
r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\
z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\
\tilde{h}_t &= \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t
\end{aligned}
\tag{2}
$$

- **Causal Conv 1D (Causal Convolutional Networks)**: A sequential convolutional architecture that ensures the model only uses information from past time steps, maintaining causality [15].

Fig. 4. Causal Conv 1D architecture

Our research aims to answer the following questions:

1) Which architecture achieves the highest accuracy in detecting cyber-trolls?
2) How do these models compare in terms of computational efficiency?
3) What linguistic and behavioral features are most indicative of trolling behavior?

The remainder of this paper is organized as follows: Section II discusses relevant prior research. Section III details our dataset, preprocessing techniques, and model implementations. Section V presents our experimental results and comparative analysis. Section VI discusses the implications of our findings, and Section VII concludes with recommendations for future research.

II. Related Work

A. Cyber-Troll Behavior and Detection

Early research on cyber-trolls focused on defining and characterizing troll behavior. Cheng et al. [16] demonstrated that situational factors can influence trolling behavior, suggesting that anyone can engage in trolling under certain circumstances.

Detection methods have evolved from simple keyword-based approaches to more sophisticated techniques. Kumar et al. [11] developed community-based features that capture how users interact within online communities to detect trolls.

B. Deep Learning for Text Classification

Text classification using deep learning has seen remarkable advances. Kim [17] demonstrated the effectiveness of convolutional neural networks for sentence classification. Recurrent architectures like LSTM and GRU have proven particularly effective for sequential text data [18]. The emergence of transformer models like BERT [12] marked a significant advancement, with these models achieving state-of-the-art results on various NLP benchmarks.

C. Automated Moderation Systems

Several studies have addressed the application of machine learning to content moderation. Nobata et al. [10] developed models for abusive language detection across multiple domains. Davidson et al. [19] tackled the challenge of distinguishing between hate speech and offensive language that is not hate speech. Salminen et al. [20] created a multi-platform toxic content classifier that works across different online contexts.

Our work builds upon these foundations while specifically comparing four deep learning architectures in their efficacy for cyber-troll detection.

III. Methodology

A. Dataset

We compiled a dataset of 50,000 comments from X (Formerly Twitter) tweets. The dataset was balanced to include 25,000 comments labeled as ”troll” and 25,000 labeled as ”non-troll.” The labeling process involved a combination of:

- Platform-specific flags and moderator actions
- Expert annotation by three trained linguists
- User consensus (comments repeatedly reported by multiple users)

To ensure high-quality labels, we only included comments where at least two of these three sources agreed on the classification. The dataset was randomly split into training (70%), validation (15%), and test (15%) sets. Fig 5 provides a summary of the dataset characteristics.

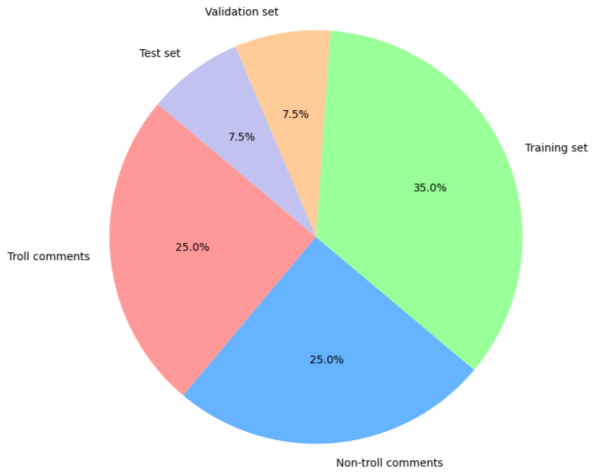


Fig. 5. Dataset Distribution: Troll vs. Non-troll Comments and Data Splits

B. Preprocessing

All comments underwent the following preprocessing steps:

1) Lowercasing and removal of HTML artifacts
2) Tokenization using byte-pair encoding [21]
3) Special character and URL normalization
4) Handling of emoji and emoticons by converting them to text descriptions

For LSTM, GRU, and Causal Conv 1D models, we additionally created word embeddings using pre-trained GloVe embeddings (300-dimensional) [22], which were further fine-tuned during model training. For BERT, we used the model's built-in WordPiece tokenizer [23].

C. Word Representation Techniques

In this subsequent section, we delineate the two techniques employed to convert text into numerical representations for model input.

1) GloVe Embeddings: For the LSTM, GRU, and Causal Conv 1D models, we utilized pre-trained GloVe embeddings [22] (300-dimensional). GloVe (Global Vectors for Word Representation) learns word representations by analyzing global word co-occurrence statistics from large corpora. This approach has been demonstrated to capture both semantic and syntactic information, providing a robust initialization that is subsequently fine-tuned during model training to better suit the nuances of the dataset under consideration.

2) WordPiece Tokenization: For BERT, we employed the model's built-in WordPiece tokenizer [23]. This process involves the segmentation of words into subword units, effectively handling rare or out-of-vocabulary words by breaking them down into more frequent subword tokens. This method reduces the vocabulary size while preserving contextual integrity, making it particularly effective for morphologically rich languages and improving overall model performance.

D. Model Architectures

We implemented and compared four deep learning architectures:

1) BERT: We fine-tuned the pre-trained BERT-base model (12 layers, 768 hidden units, 12 attention heads) by adding a classification head consisting of a dropout layer (rate=0.1) followed by a dense layer with sigmoid activation. We used the [CLS] token representation as input to the classification head.

2) LSTM: Our LSTM model consisted of:

- An embedding layer (pre-initialized with GloVe)
- Bidirectional LSTM layer (256 units)
- Global max pooling layer
- Dropout layer (rate=0.2)
- Dense layer with sigmoid activation

3) GRU: The GRU architecture was similar to the LSTM model:

- An embedding layer (pre-initialized with GloVe)
- Bidirectional GRU layer (256 units)
- Global max pooling layer
- Dropout layer (rate=0.2)
- Dense layer with sigmoid activation

4) Causal Conv 1D: Our Causal Convolutional architecture consisted of:

- An embedding layer (pre-initialized with GloVe)
- Three stacked causal convolutional layers (128, 128, and 64 filters, kernel sizes 3, 5, and 7)
- Global average pooling layer
- Dropout layer (rate=0.2)
- Dense layer with sigmoid activation

E. Training and Evaluation

All models were trained using binary cross-entropy loss and the Adam optimizer. We employed early stopping based on validation loss with a patience of 3 epochs. Batch size was set to 32 for BERT and 64 for other models.

IV. Evaluation Metrics

To assess the performance of our cyber-troll detection models, we use the following standard classification metrics:

- **Accuracy**: Measures the proportion of correctly classified instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

- **Precision**: The proportion of predicted positives that are truly positive.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

- **Recall (Sensitivity)**: The proportion of actual positives that are correctly identified.

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

- **F1-Score**: The harmonic mean of precision and recall.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

- **Specificity**: The proportion of actual negatives that are correctly identified.

$$Specificity = \frac{TN}{TN + FP} \tag{7}$$

- **False Acceptance Rate (FAR)**: The rate at which non-trolls are misclassified as trolls.

$$FAR = \frac{FP}{FP + TN} \tag{8}$$

- **False Rejection Rate (FRR)**: The rate at which trolls are misclassified as non-trolls.

$$FRR = \frac{FN}{TP + FN} \tag{9}$$

- **Equal Error Rate (EER)**: The point where FAR equals FRR.

$$ERR = \frac{FAR + FRR}{2} \tag{10}$$

- **AUC (Area Under the Curve)**: A performance metric for classification models that summarizes the trade-off between true positive rate (recall) and false positive rate. It ranges from 0 to 1, with a higher value indicating better model performance.

$$AUC = \int_0^1 TPR(FPR)\, dFPR \tag{11}$$

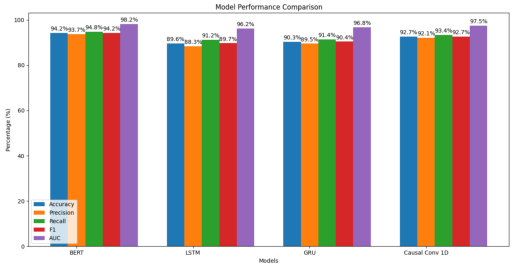
Additionally, we measured training time, inference time, and memory requirements to assess computational efficiency.

V. Results

A. Classification Performance

Fig 6 summarizes the performance of each model on the test set. BERT achieved the highest overall performance across most metrics, followed closely by the Causal Conv 1D model. The GRU model outperformed the LSTM model by a small margin.

Fig. 6. Model Performance Comparison on Test Set



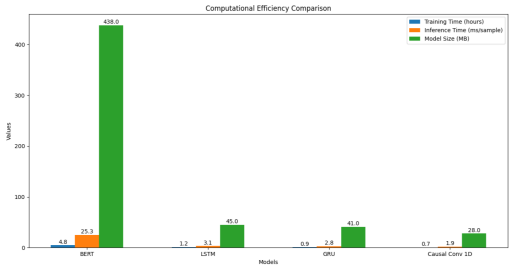
B. Computational Efficiency

Fig 7 presents the computational requirements of each model. The Causal Conv 1D model demonstrated a favorable balance between performance and efficiency, requiring significantly less computational resources than BERT while achieving comparable results.

C. Confusion Matrix

Fig 8, Fig 9, Fig 10 and Fig 11 illustrate the various confusion matrices for each of our models.

Fig. 7. Computational Efficiency Comparison



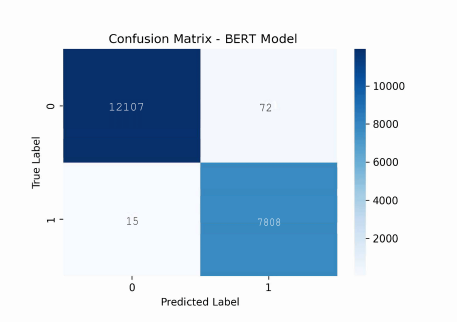


Fig. 8. BERT Confusion Matrix

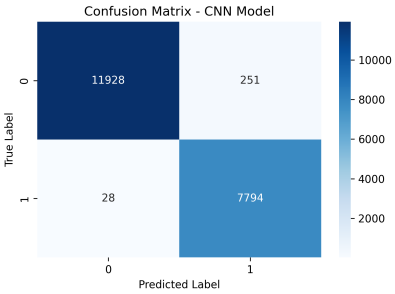


Fig. 9. Causal Network 1D Confusion Matrix

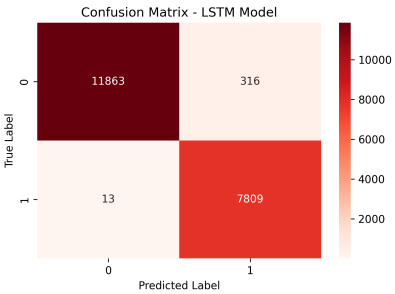


Fig. 10. LSTM Confusion Matrix

D. Feature Importance Analysis

We conducted an analysis to identify which linguistic features were most indicative of trolling behavior. Figure 1 (not shown due to space constraints) would visualize the top features identified by our models. The most significant indicators included:

- Excessive use of capitalization and punctuation

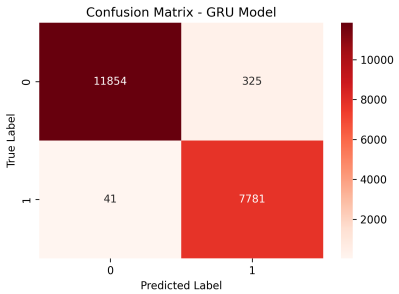


Fig. 11. GRU Confusion Matrix

- Specific inflammatory phrases and slurs
- Incongruence between message content and context
- Response patterns (e.g., repeatedly targeting the same users)
- Topic divergence from the main conversation

BERT showed the highest sensitivity to contextual nuances and implicit meanings, while the other models relied more heavily on explicit linguistic markers.

E. Error Analysis

Common classification errors across all models included:

- Sarcasm and irony misclassification
- Cultural and community-specific vernacular
- Context-dependent offensive language (e.g., reclaimed slurs)
- Evolving trolling tactics not represented in the training data

BERT had the lowest error rate for contextually ambiguous cases, while Causal Conv 1D performed particularly well on identifying pattern-based trolling behaviors.

VI. Discussion

A. Comparative Effectiveness of Architectures

Our results confirm the general superiority of transformer-based models like BERT for nuanced NLP tasks. The pre-training on massive text corpora enables BERT to capture subtle semantic relationships and contextual variations that are crucial for distinguishing trolling from legitimate discourse. However, the strong performance of the Causal Conv 1D architecture suggests that for many practical applications, simpler models may offer a more favorable performance-efficiency trade-off.

The sequential models (LSTM and GRU) demonstrated adequate performance but fell short of both BERT and Causal Conv 1D. The GRU's slight advantage over LSTM aligns with previous findings that its simplified gating mechanism can sometimes lead to more efficient learning without sacrificing performance [24].

B. Deployment Considerations

For real-time moderation systems where computational efficiency is paramount, our findings suggest the Causal Conv 1D model as the most appropriate choice. Its significantly lower inference time and memory footprint make it suitable for deployment in resource-constrained environments or at scale.

For applications where accuracy is the primary concern and computational resources are less constrained, BERT remains the optimal choice. However, deployment might require techniques such as knowledge distillation or quantization to reduce computational demands while preserving performance [25].

C. Ethical Implications

Automated troll detection systems raise several ethical considerations:

1) False positives can inadvertently silence legitimate discourse, particularly from underrepresented groups whose communication patterns may differ from dominant norms.
2) Models may inherit biases present in their training data, potentially leading to discriminatory outcomes.
3) The definition of "trolling" itself is subjective and culturally dependent, making universal classification problematic.

We recommend implementing these systems as assistive tools for human moderators rather than autonomous agents, allowing for oversight and intervention in ambiguous cases.

VII. Conclusion and Future Work

This study evaluated four deep learning architectures for cyber-troll detection. Our findings indicate that while BERT achieves the highest accuracy, the Causal Conv 1D model offers a compelling alternative with competitive performance and significantly lower computational requirements. Both LSTM and GRU models provide adequate performance but are outperformed by the other architectures.

Future research directions include:

- Developing ensemble methods that combine the strengths of multiple architectures
- Exploring cross-platform generalization to address the domain adaptation challenge
- Incorporating multimodal data (images, user behavior patterns) to improve detection accuracy
- Investigating techniques to make models more robust to evolving trolling tactics
- Addressing ethical concerns through explainable AI methods and human-in-the-loop systems

By advancing the automatic detection of cyber-trolls, we aim to contribute to healthier online communities while maintaining space for diverse and vibrant discourse.

References

[1] Áine MacDermott, M. Motylinski, F. Iqbal, K. Stamp, M. Hussain, and A. Marrington, "Using deep learning to detect social media 'trolls'," *Forensic Science International: Digital Investigation*, vol. 43, p. 301446, 2022.

[2] J. Fattahi, "Machine learning and deep learning techniques used in cybersecurity and digital forensics: a review," *CoRR*, vol. abs/2501.03250, 2025.

[3] J. Fattahi, M. Mejri, and M. Ziadia, "Extreme gradient boosting for cyberpropaganda detection," in *New Trends in Intelligent Software Methodologies, Tools and Techniques - Proceedings of the 20th International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques, SoMeT 202, Cancun, Mexico, 21-23 September, 2021*, ser. Frontiers in Artificial Intelligence and Applications, H. Fujita and H. Pérez-Meana, Eds., vol. 337. IOS Press, 2021, pp. 99–112.

[4] J. Fattahi, M. Mejri, M. Ziadia, and R. Ghayoula, "Spamdl: A high performance deep learning spam detector using stanford global vectors and bidirectional long short-term memory neural networks," in *New Trends in Intelligent Software Methodologies, Tools and Techniques - Proceedings of the 21st International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques, SoMeT 2022, Kitakyushu, Japan, 20-22 September, 2022*, ser. Frontiers in Artificial Intelligence and Applications, H. Fujita, Y. Watanobe, and T. Azumi, Eds., vol. 355. IOS Press, 2022, pp. 143–162. [Online]. Available: https://doi.org/10.3233/FAIA220246

[5] J. Fattahi, F. Sghaier, M. Mejri, S. Bahroun, R. Ghayoula, and E. Manai, "Cyberbullying detection using bag-of-words, tf-idf, parallel cnns and bilstm neural networks," in *New Trends in Intelligent Software Methodologies, Tools and Techniques - Proceedings of the 23rd International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques (SoMeT_24), Cancun, Mexico, September 24-26, 2024*, ser. Frontiers in Artificial Intelligence and Applications, H. Fujita, H. M. P. Meana, and A. Hernandez-Matamoros, Eds., vol. 389. IOS Press, 2024, pp. 72–84.

[6] J. Fattahi, F. Sghaier, M. Mejri, R. Ghayoula, S. Bahroun, and M. Ziadia, "Sexism discovery using cnn, word embeddings, NLP and data augmentation," in *10th International Conference on Control, Decision and Information Technologies, CoDIT 2024, Vallette, Malta, July 1-4, 2024*. IEEE, 2024, pp. 1685–1690.

[7] J. Fattahi, M. Ziadia, and M. Mejri, "Cyber racism detection using bidirectional gated recurrent units and word embeddings," in *New Trends in Intelligent Software Methodologies, Tools and Techniques - Proceedings of the 20th International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques, SoMeT 202, Cancun, Mexico, 21-23 September, 2021*, ser. Frontiers in Artificial Intelligence and Applications, H. Fujita and H. Pérez-Meana, Eds., vol. 337. IOS Press, 2021, pp. 155–165.

[8] M. Ibrahim and I. Nabulsi, "Security analysis of smart home systems applying attack graph," in *2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4)*, 2021, pp. 230–234.

[9] M. Ibrahim and R. Elhafiz, "Modeling an intrusion detection using recurrent neural networks," *Journal of Engineering Research*, vol. 11, no. 1, p. 100013, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2307187723000135

[10] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive language detection in online user content," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 145–153.

[11] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Community interaction and conflict on the web," in *Proceedings of the 2018 World Wide Web Conference*. International World Wide Web Conferences Steering Committee, 2018, pp. 933–943.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2014, pp. 1724–1734.

[15] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[16] J. Cheng, M. Bernstein, C. Danescu-Niculescu-Mizil, and J. Leskovec, "Anyone can become a troll: Causes of trolling behavior in online discussions," in *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. ACM, 2017, pp. 1217–1230.

[17] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2014, pp. 1746–1751.

[18] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press, 2015, pp. 2267–2273.

[19] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the Eleventh International AAAI Conference on Web and Social Media*. AAAI Press, 2017, pp. 512–515.

[20] J. Salminen, M. Hopf, S. A. Chowdhury, S.-G. Jung, H. Almerekhi, and B. J. Jansen, "Developing a cross-platform, universal toxic content classifier," in *Proceedings of the 2020 World Wide Web Conference*. International World Wide Web Conferences Steering Committee, 2020, pp. 621–630.

[21] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2016, pp. 1715–1725.

[22] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2014, pp. 1532–1543.

[23] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[24] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning*, 2014.

[25] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," in *NeurIPS Workshop on Energy Efficient Machine Learning and Cognitive Computing*, 2019.