Decoding Cyberbullying on Social Media: A Machine Learning Exploration

Aisha Saeid
School of Computer Science
and Electronic Engineering
University of Surrey
Guildford, United Kingdom
a.saeid@surrey.ac.uk

Diptesh Kanojia
School of Computer Science
and Electronic Engineering
University of Surrey
Guildford, United Kingdom
d.kanojia@surrey.ac.uk

Ferrante Neri
School of Computer Science
and Electronic Engineering
University of Surrey
Guildford, United Kingdom
f.neri@surrey.ac.uk 0000-0002-6100-6532

Abstract—**Social media, a vast platform for communication and entertainment, unfortunately, is an ideal breeding ground for cyberbullying. While most common among teenagers, it also affects other demographics. Despite strict zero-tolerance policies on social media, the elusive nature of cyberbullying persists. Simple word searches are insufficient, leading to the exploration of Natural Language Processing (NLP) to detect and classify cyberbullying. This study balances result accuracy with model simplicity, crucial for an effective detector. Quick identification of offensive content is essential to combat cyberbullying. The ever-changing slang and trends require an easily updatable detector. Using a cyberbullying dataset from real tweets on X (formerly Twitter), this study initially applies traditional machine learning algorithms, including Logistic Regression, Support Vector Machines, Decision Trees, and Random Forests. The investigation then moves to Transformers-based autoencoders from the BERT family, including sentence-transformers. However, these models require significant memory and disk space due to their large number of training parameters. The study focuses on the efficiency of cyberbullying detectors using character-level language models based on the bidirectional long-short-term memory (BiLSTM) neural architecture. Our experiments demonstrate that these detectors offer comparable performance and provide a practical option for real-world deployment.**

Index Terms—**Cyberbullying, natural language processing, text classification, machine learning**

I. Introduction

In the digital age, social media platforms offer unparalleled communication opportunities but also enable the alarming rise of cyberbullying [1]. As people increasingly express themselves online, cyberbullying, defined as using electronic communication to harass or intimidate, threatens digital well-being [2]. The intersection of technology and empathy provides hope [3]. The term ’bully’ originates from the 1500s, involving an intimidator and a victim [4]. Bullying, a way to gain power or superiority, is considered an instinctual survival trait, with various tactics learned from early ages persisting into adulthood, impacting others negatively [5]. With the evolution of technology and the Internet’s ubiquity, Web 2.0 platforms facilitate social media growth. Datareportal’s April 2023 Global Overview [6] shows over half the world (4.80 billion) uses social media, with around 150 million new users in the past year. The shift to cyberbullying has been facilitated by the distance and anonymity social media offers. Despite being a complex social phenomenon, comprehensive classification of cyberbullying remains an ongoing challenge [7]. The study in [8] identifies the following main categories of cyberbullying:

Trolling – Posting provocative messages to bait the victim for an emotional response.

Harassment – Sending threatening or harassing messages.

Cyberstalking – Stalking others’ information for false accusations, monitoring, identity theft, threats, or data destruction.

Masquerade – Using a fake profile to cyberbully under another identity.

Flaming – Using vulgar language to provoke someone

Denigration – Unfairly criticising someone online by spreading damaging gossip and rumours.

Apart from classifying cyberbullying by attack types, it can also be categorised based on content, which is crucial for identifying the crime and determining appropriate countermeasures. Examples include attacks on religious, ethnic, or age groups. These categories evolve with society, making cyberbullying a complex phenomenon and cyberbullying classification a difficult task. To address this, labelled datasets have been proposed [9] with offensive content-based categorisations.

The rise of machine learning and natural language processing (NLP) has enabled automated detection and classification of cyberbullying in recent decades. Early methods [10] focused on using social information and textual characteristics to identify cyberbullying. As machine learning advanced, more sophisticated systems were developed. For instance, [11] used the Levenshtein algorithm for word detection and Naive Bayes for classification, but faced computational challenges. [12] employed fuzzy SVM for data augmentation and semi-supervised tweet classification. Meanwhile, [13] proposed collaborative tweet analysis for improved efficiency, outperforming sequential methods in 7 of 15 cases. Building on this, [14] introduced a probabilistic fusion paradigm considering confidence ratings and various social and textual factors. With the rise of deep neural networks, many systems using deep learning for cyberbullying detection and classification have emerged. For example, [15] introduced a method that enhances Twitter cy-

This work is funded by the Jiangsu Distinguished Professor programme.

979-8-3503-5409-6/24/$31.00 ©2024 IEEE
DOI 10.1109/CAI59869.2024.00084

berbullying detection by using word vectors to preserve tweet semantics, bypassing traditional feature extraction. An optimisation algorithm fine-tuned the parameters of a convolutional neural network (CNN) for optimal results. Another study, [16], employed a multi-layer CNN. Additionally, modern neural architectures have been explored for cyberbullying detection across different languages [17]–[19].

To improve multi-classification performance in cyberbullying detection, recent systems often use hybrid algorithms. A study in [19] explored eleven classification algorithms and seven feature extraction methods across two datasets, highlighting the effectiveness of attention models and bidirectional neural networks. Logistic regression was the most efficient traditional classifier, and Term Frequency-Inverse Document Frequency consistently achieved high accuracy. However, deep neural networks generally outperform traditional methods. Another study [20] used Large Language Models (LLMs) and traditional classifiers to identify and classify two prominent cyberbullying types—personal assaults and hate speech—on platforms like Twitter and Wikipedia. Building on this research, the study in [7] introduces and compares two cyberbullying detection architectures. The first is an ensemble model using a CNN for feature extraction and an SVM classifier. The second employs a pre-trained BERT model. A systematic review [21] highlights persistent challenges in leveraging machine learning for cyberbullying detection, including multi-language platforms and underexplored areas in unsupervised learning. The complex and evolving nature of cyberbullying continues to challenge even advanced machine learning algorithms. An open question remains: whether traditional machine learning, deep learning like Transformers-based models, or pre-trained LLMs offer the best balance between reliability and simplicity.

In light of these considerations, the present study presents a comparative analysis of four traditional machine learning models and four Transformers-based models for classifying tweets into six cyberbullying classes. We also investigate character-level language models, which have shown comparable performance to Transformers in tasks like Named Entity Recognition [22]. We approach cyberbullying detection as a classification problem using a balanced dataset and evaluate multiple language models. Our analysis shows that character-level language models are efficient due to their bidirectional context capture and comparable performance to larger Transformer-based models, which increase computational costs. We report accuracy, macro- and weighted-F1 scores, and apply ensemble learning to assess the collective impact of diverse algorithms.

The remainder of this article is organised as follows. Section II details the dataset and the cyberbullying classification system. Section III presents our comparative results, and Section IV concludes and discusses potential future work.

II. Cyberbullying Classification

This section outlines the methodology to detect cyberbullying tweets from X (formerly Twitter) using a mixed-method approach combining qualitative and quantitative analysis. We use a publicly available dataset from the Kaggle platform for cyberbullying detection. This dataset provides $51,718$ tweets categorized into various forms of cyberbullying, and non-cyberbullying tweet instances. We use different approaches, such as traditional machine learning algorithms combined with TF-IDF and Count vectorization, fine-tuning pre-trained Transformers-based models, and pre-trained character-level language models. We conduct classification experiments to identify the best approach and comprehensively evaluate various language models on the same dataset. We aim to identify the most efficient approaches and train robust models for cyberbullying detection.

The dataset in [9] presents a slight imbalance in data with Religion 16%, Age 16%, Gender 16%, Ethnicity 15%, other cyberbulling 15%, and not cyberbullying 22%. ML methods work under the assumption of a balanced distribution of data, which is not the case with real-world data. Thus, we used the `RandomUnderSampler` operator to rebalance the dataset. We further divide the data into training (70%), development (15%), and test (15%) sets.

For the ML algorithms, we use `TfIDFVectorizer`, removing English stop words, with a 1-gram range and a vocabulary size of $10,000$. These vectors are used by traditional ML algorithms in training with labels in a supervised setting. Pre-trained language models based on Transformers use their own tokenizers to split raw text into tokens at the subword level, based on a set vocabulary from the pre-training corpus. In contrast, character-level language models use a simple character-level vocabulary from the monolingual corpus used for pre-training, avoiding the need for complex tokenization like Transformers.

A. Experiment Setup

In our experiments, we group methods from the same paradigm into three settings. The first explores traditional machine learning-based methods. The second focuses on Transformer-based pre-trained language models, and the third uses Flair's character-level language models. We limit our investigation to pre-trained language models and exclude autoregressive decoder-based LLMs, which generate text rather than class labels.

1) Traditional Machine Learning algorithms: Initially, we employed four traditional machine learning algorithms for classification: Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF). SVM is a robust classifier known for its sparse approach, kernel method, and maximum margin separation [23]. LR is a standard algorithm used for classification. DT uses a recursive or iterative division of the instance space to construct a rooted tree with nodes and leaves [24]. RF, capable of handling both regression and multiclass classification, offers a built-in estimation of generalisation errors and allows for measurement of variable importance [25].

2) Pre-trained Transformers-based Langauge Models: In this setting, we employ attention-based Transformer models, also known as autoencoders, which contextualise token and sentence representations within sentences and snippets [26]. We choose Transformer encoders for their superior performance and shorter training times compared to recurrent networks [27]. BERT, or Bidirectional Encoder Representations from Transformers, is a pre-trained language model designed to create bidirectional representations from unlabelled text, enabling the construction of state-of-the-art models with minimal modifications [28]. Given the cost and complexity of operating large-scale pre-trained models, DistilBERT offers a smaller, 60% faster, and more efficient alternative, retaining 97% of BERT's language understanding capabilities [29].

RoBERTa is a refined version of BERT designed to address its limitations, such as cost and restricted tuning. It improves upon BERT by removing the next sentence prediction objective, using longer sequences, and adapting the masking pattern during training [30]. Albert-base, on the other hand, enhances performance through two parameter reduction techniques: factorised embedding parameterisation, which splits the large vocabulary embedding matrix to allow for increased hidden size without adding parameters, and cross-layer parameter sharing, which prevents parameter growth with network depth [31]. We also employ Sentence-BERT (SBERT), a pre-trained network using Siamese and triplet structures to generate semantically meaningful sentence embeddings compared using cosine similarity [32]. It offers faster results than BERT or RoBERTa for finding related or semantically similar sentences. However, the speedup varies based on factors like the task, dataset size, and computational resources. While SBERT is quicker, RoBERTa may offer better accuracy in certain tasks [33]. Thus, the model choice depends on project needs, prioritising speed or accuracy. We use the all-mpnet-base-v2 model from SBERT for its reported performance.

3) FLAIR: Fast & Lightweight Analysis and Identification of Named Entities and Linguistic Relations: Akbik et al. [34] introduced a character-level language model in the FLAIR framework (https://github.com/flairNLP/flair). These Long Short Term Memory (LSTM) networks rival Transformers in performance [35]. Durining pre-training, each LSTM predicts a 256-character sequence, addressing long-term dependency issues in Recurrent Neural Networks (RNNs). In this study, we use the Bidirectional Long Short Term Memory (BiLSTM) architecture from the multi-X and news-X-fast pre-trained models. The multi-X model supports multiple languages, and news-X-fast is trained on general domain English data. FLAIR provides Pooled embeddings, enhancing token-level information during training. We also experiment with Stacked embeddings, combining Transformers-based and FLAIR-based character-level model embeddings. We choose the best Transformers-based model to stack with both multi-X and news-X-fast to assess performance. These models capture the context of each character, enabling a nuanced understanding of language. Pre-trained models like multi-X and news-X-fast conserve computational resources by leveraging existing learned representations rather than training from scratch, with significantly less storage and processing time compared to Transformers-based models.

B. Training and Hyperparameters

In the first setting, we use traditional ML algorithms (SVM, LR, DT, and RF). LR is trained with the regularisation parameter C set to 2.0, and SVM uses a linear kernel with C at 1.0. Training these ML models on the vectorised data finished within 30 minutes each. For ensemble learning, we employ a majority voting approach. We use HuggingFace APIs to access pre-trained language models and fine-tune each autoencoder model, conducting experiments on a GPU with 16 GB VRAM. We used a batch size of 16 and weight decay of 0.01. Fine-tuning runs with DistilBERT, Roberta-base, Albert-base-v2, and All-mpnet-v2 took about 3 hours each, totalling 12 hours. Using pre-trained character-level embeddings, we trained an LSTM classifier with a 512 hidden layer size, a learning rate of 5e-05, and a mini-batch size of 64, training over 50 epochs with early stopping. We used the same hyperparameters with the better-performing character-level model for Pooled embeddings. The mini-batch size was reduced to 16 for Stacked embeddings due to memory constraints.

III. Results and Analysis

To evaluate the effectiveness of the proposed models for cyberbullying detection, we use the F1 score as the primary metric. For each experiment, we discuss the accuracy, macro averaged F1 (F1_{macro}) and weighted average F1 ($\text{F1}_{weighted}$).

Table I displays results achieved by the models under consideration. LR performs best with a macro-F1 of 0.84. While a majority voting ensemble matches this, LR is preferable for individual performance. SVM is close but has a lower weighted average and accuracy. LR and ensemble models show the highest results and require further monitoring with more data. DistilBERT, Roberta-base, and Albert-base-v2 achieved top accuracy scores of 0.85, 0.86, and 0.85, respectively. The all-mpnet-base-v2 model performed lowest with 0.83 accuracy, 0.84 macro average, and 0.83 weighted average. Combining the top three Transformers-based models in an ensemble yielded the highest accuracy of 0.86 among both ML and Transformers-based models.

Character-level FLAIR framework models, multi-X and news-X-fast, show strong performance comparable to Transformers (Table I). Individually, they may not surpass some traditional machine learning algorithms. While TF-IDF doesn't capture context, character-level embeddings do. When token-level information is pooled and concatenated, their performance matches Transformers. FLAIR Stacked embeddings of DistilBERT with multi-X achieved 0.83 accuracy, with macro and weighted averages of 0.84 and 0.83. Overall, Pooled Flair embedding with multi-X exhibits the highest accuracy and weighted average. Model sizes differ by a factor of five between character-level models and Transformers-based models. Inference time on the test set varies by a factor of two. Stacked models are around twice the size of Transformer-based models without justified performance.

TABLE I

QUANTITATIVE RESULTS OF CYBER-BULLYING DETECTION

Traditional ML Model	**Accuracy**	**F1**$_{macro}$	**F1**$_{Weighted}$
Logistic Regression (LR)	0.83	0.84	0.83
Support Vector Machine (SVM)	0.82	0.84	0.82
Decision Tree (DT)	0.79	0.81	0.79
Random Forest (RF)	0.78	0.80	0.78
Ensemble Learning	0.83	0.84	0.83
Transformer-based Language Model	**Accuracy**	**F1**$_{macro}$	**F1**$_{Weighted}$
DistilBERT	0.85	0.86	0.85
Roberta-base	0.85	0.86	0.85
Albert-base-v2	0.85	0.86	0.85
All-mpnet-base-v2	0.83	0.84	0.83
Ensemble Learning	0.86	0.87	0.86
FLAIR Model	**Accuracy**	**F1**$_{macro}$	**F1**$_{Weighted}$
Flair Embeddings			
multi-X	0.82	0.83	0.82
news-X-fast	0.81	0.83	0.82
Pooled Flair Embeddings			
multi-X	0.84	0.85	0.84
Stacked Embeddings			
multi-X and DistilBERT	0.83	0.84	0.83
news-X-fast and DistilBERT	0.83	0.84	0.82

IV. CONCLUSION AND FUTURE WORK

In this study, we explore cyberbullying detection on social media using a dataset from platform X with cyberbullying labels. We compare traditional ML algorithms, fine-tuned Transformers, and character-level BiLSTM models. Our experiments show that character-level models perform similarly to Transformers, and pooling token-level information enhances classification. We also evaluate model size, training, and inference times, suggesting BiLSTM models are suitable for real-time deployment. Fast inference is vital for practical deployment, as Transformers require expensive infrastructure. Therefore, these efficient models could be preferable for cyberbullying detection. However, since these models aren't pre-trained on social media data, additional pre-training on relevant data may be needed for improved performance.

REFERENCES

[1] E. Whittaker and R. M. Kowalski, "Cyberbullying via social media," *Journal of School Violence*, vol. 14, no. 1, pp. 11–29, 2015.

[2] G. W. Giumetti and R. M. Kowalski, "Cyberbullying via social media and well-being," *Curr. Opinion in Psychology*, vol. 45, p. 101314, 2022.

[3] M. F. López-Vizcaíno, F. J. Nóvoa, V. Carneiro, and F. Cacheda, "Early detection of cyberbullying on social media networks," *Future Generation Computer Systems*, vol. 118, pp. 219–229, 2021.

[4] E. Kraft and J. Wang, "An exploratory study of the cyberbullying and cyberstalking experiences and factors related to victimization of students at a public liberal arts college," *Int. J. Technoethics*, vol. 1, no. 4, pp. 74–91, 2010.

[5] N. M. Zainudin, K. H. Zainal, N. A. Hasbullah, N. A. Wahab, and S. Ramli, "A review on cyberbullying in malaysia from digital forensic perspective," in *2016 ICICTM*, 2016, pp. 246–250.

[6] "Datareportal April 2023 global overview," https://datareportal.com/reports/digital-2023-april-global-statshot, 2023.

[7] H. Saini, H. Mehra, R. Rani, G. Jaiswal, A. Sharma, and A. Dev, "Enhancing cyberbullying detection: a comparative study of ensemble cnn–svm and bert models," *Social Network Analysis and Mining*, vol. 14, no. 1, 2024.

[8] D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, "Detection of harassment on web 2.0," in *Proc. of Content Analysis in the WEB 2.0*, 2009.

[9] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on twitter," in *Proc. 2017 ACM on Web Science Conference*, 2017, p. 13–22.

[10] K. Reynolds, A. Kontostathis, and L. Edwards, "Using machine learning to detect cyberbullying," in *ICMLA*, vol. 2, 2011, pp. 241–244.

[11] B. S. Nandhini and J. I. Sheeba, "Cyberbullying detection and classification using information retrieval algorithm," in *ACM ICARCSET*, 2015.

[12] V. Nahar, S. Al-Maskari, X. Li, and C. Pang, "Semi-supervised learning for cyberbullying detection in social networks," in *Databases Theory and Applications: 25th ADC*. Springer, 2014, pp. 160–171.

[13] A. Mangaonkar, A. Hayrapetian, and R. Raje, "Collaborative detection of cyberbullying behavior in twitter data," in *EIT*, 2015, pp. 611–616.

[14] V. K. Singh, Q. Huang, and P. K. Atrey, "Cyberbullying detection using probabilistic socio-textual information fusion," in *2016 IEEE/ACM ASONAM*, 2016, pp. 884–887.

[15] M. A. Al-Ajlan and M. Ykhlef, "Optimized twitter cyberbullying detection based on deep learning," in *21st Saudi Computer Society NCC*, 2018, pp. 1–5.

[16] V. Banerjee, J. Telavane, P. Gaikwad, and P. Vartak, "Detection of cyberbullying using deep neural network," in *ICACCS*, 2019, pp. 604–607.

[17] A. Dewani, M. A. Memon, and S. Bhatti, "Cyberbullying detection: advanced preprocessing techniques & deep learning architecture for roman urdu data," *Journal of big data*, vol. 8, no. 1, p. 160, 2021.

[18] M. H. Obaid, S. K. Guirguis, and S. M. Elkaffas, "Cyberbullying detection and severity determination model," *IEEE Access*, 2023.

[19] C. Raj, A. Agarwal, G. Bharathy, B. Narayan, and M. Prasad, "Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques," *Electr.*, vol. 10, no. 22, 2021.

[20] V. Jain, V. Kumar, V. Pal, and D. K. Vishwakarma, "Detection of cyberbullying on social media using machine learning," in *5th ICCMC*, 2021, pp. 1091–1096.

[21] V. Balakrisnan and M. Kaity, "Cyberbullying detection and machine learning: a systematic literature review," *Artif Intell Rev*, vol. 56, no. 1, pp. 1375–1416, 2023.

[22] A. Roy, "Recent trends in named entity recognition (ner)," *arXiv preprint arXiv:2101.11420*, 2021.

[23] M. Awad, R. Khanna, M. Awad, and R. Khanna, "Support vector regression," *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*, pp. 67–80, 2015.

[24] O. Maimon and L. Rokach, "Introduction to knowledge discovery in databases," in *Data mining and knowledge discovery handbook*. Springer, 2005, pp. 1–17.

[25] A. Haj-Ali, N. K. Ahmed, T. Willke, Y. S. Shao, K. Asanovic, and I. Stoica, "Neurovectorizer: End-to-end vectorization with deep reinforcement learning," in *Proc. 18th ACM/IEEE ISCGO*, 2020, pp. 242–255.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[27] S. Somasundaran *et al.*, "Two-level transformer and auxiliary coherence modeling for improved text segmentation," in *Proceedings of the AAAI CAI*, vol. 34, no. 05, 2020, pp. 7797–7804.

[28] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[29] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[30] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[31] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," *arXiv preprint arXiv:1909.11942*, 2019.

[32] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[33] D. Cortiz, "Exploring transformers in emotion recognition: a comparison of bert, distillbert, roberta, xlnet and electra," *arXiv preprint arXiv:2104.02041*, 2021.

[34] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *Proc. 27th ICCL*, 2018, pp. 1638–1649.

[35] D. Xu, E. Laparra, and S. Bethard, "Pre-trained contextualized character embeddings lead to major improvements in time normalization: A detailed analysis," in *Proc.SEM*, 2019, pp. 68–74.