International Journal of Advances in Scientific Research and Engineering (ijasre)

E-ISSN : 2454-8006

DOI: 10.31695/IJASRE.2024.3.3

Volume 10, Issue 3
March - 2024

Cyberbullying Messages Detection Using Machine Learning and Deep Learning

Jinan Redha Mutar

Department of Computer Science

Collage of Education, Mustansiriyah University

Baghdad, Iraq

ABSTRACT

Cyberbullying has emerged as a significant concern in contemporary times, particularly due to its severe consequences, especially for children. In this paper, we propose an innovative machine learning-based approach aimed at accurately detecting cyberbullying messages and mitigating their harmful effects. The primary objectives of our research were twofold: developing a model capable of precisely identifying cyberbullying messages while distinguishing them from regular messages. To achieve this, we utilized a dataset of social media messages, labeled as normal, offensive, or hate messages. We adapted this dataset for binary classification, differentiating between cyberbullying and non-bullying messages. Our approach involved two distinct methods: firstly, utilizing Term Frequency-Inverse Document Frequency (TF-IDF) for traditional machine learning algorithms, and secondly, embedding texts for deep learning algorithms. We employed a total of 15 classifiers and performes a comprehensive comparison. The most successful algorithms from the first method were combined into a voting classifier, which demonstrated the highest accuracy of 96.5% during testing. Additionally, we assessed the impact of Recursive Feature Elimination with Cross-Validation (RFECV) on the model's performance and compared it with our baseline approach. Although the results exhibited slight fluctuations, the voting classifier consistently outperformed others with 96.6% accuracy. Our findings underline the effectiveness of the voting classifier based on machine learning algorithms, which delivered the most promising results. This approach holds the potential to be implemented in social media platforms or chat applications, serving as a valuable tool in the ongoing efforts to combat cyberbullying.

Key Words: Cyberbullying, Machine Learning, Natural Language Processing, Recursive Feature Elimination, Text Classification.

1. INTRODUCTION

The advent of the internet and social networks has revolutionized the way people communicate, offering easy and anonymous means of interaction. However, this newfound freedom of expression has also given rise to a disturbing phenomenon known as cyberbullying, which we define as the malicious use of digital communication tools to harass, humiliate, or threaten individuals or groups [1]. Cyberbullying takes various forms, including the dissemination of pictures, videos, voice recordings, and text messages. While it was once considered a low-prevalence issue [2], the prevalence of cyberbullying has surged over the years, largely due to increased exposure on social media platforms [3]. Today, cyberbullying affects individuals of all age groups, either as victims or perpetrators. Unfortunately, social networks often fall short in providing adequate protection, resulting in severe consequences such as stress and enduring psychological effects like anxiety, depression, and tragically, even suicide [4]. Children are particularly vulnerable to these harmful effects [5 - 7] and the lack of effective prevention measures only exacerbates the problem [8 - 10]. These factors underscore the urgent need for tools capable of detecting and preventing cyberbullying. The vast volume of messages exchanged on social media has made cyberbullying harder to combat, as manual analysis becomes nearly impossible. Machine learning, with its ability to analyze massive amounts of data, offers a promising approach to addressing this problem.

The primary objective of this paper is to propose a machine learning-based approach for accurately identifying cyberbullying messages. Our approach relies on a dataset of social media messages, encompassing hate speech, offensive content, and normal messages, as provided by Davidson et al. [11] and annotated by CrowdFlower. To account for both offensive and hate messages as forms of cyberbullying, we merged them into a single class.

Our approach comprises two key phases: the first involves traditional classification techniques, while the second employs Recursive Feature Elimination with Cross-Validation (RFECV). In both phases, we explore two methods: one based on TF-IDF (Term Frequency-Inverse Document Frequency) for traditional machine learning algorithms and another utilizing text embedding for deep learning. For the TF-IDF phase, we employ a range of classifiers, including Logistic Regression, Decision Tree, Extra Trees, Random Forest, Support Vector Machine, K-Nearest Neighbors, Naive Bayes, Ada Boost, Gradient Boosting, Cat Boost, Extreme Gradient Boosting, Light Gradient Boosting, and Soft Voting. In the second phase involving text embedding, we leverage Convolutional Neural Networks and Bidirectional Long Short-Term Memory networks (BiLSTM). To evaluate the performance of these models, we employ metrics such as Receiving Operator Characteristic (ROC) curves, accuracy, precision, recall, and the F1-score.

The remainder of this paper is organized into several sections: Section 2 reviews related work in the field, Section 3 describes our dataset and outlines our approach to tackling this problem, Section 4 presents our results and provides an in-depth analysis, and finally, in Section 5, we conclude our study and discuss potential avenues for future research.

2. RELATED WORK

Numerous studies have emerged in recent years, focusing on effectively detecting cyberbullying messages through various machine learning techniques. Different approaches have been explored, with a particular emphasis on employing machine learning algorithms for classification. Prior research has delved into traditional machine learning algorithms for cyberbullying detection. For instance, Islam et al. [12] demonstrated the superiority of TF-IDF over Bag-of-Words (BoW) in achieving better results. Among the algorithms compared, Support Vector Machine (SVM) outperformed others on two datasets of social network messages in detecting cyberbullying. Similarly, Muneer et al. [13] evaluated several classifiers using TF-IDF and Word2Vec, with the Logistic Regression classifier displaying the highest accuracy of 90.57%.

Nandhini et al. [14] proposed a model based on Naive Bayes, achieving an accuracy of 91% on one of their datasets. Hani et al. [15] compared SVM and Neural Network (NN) classifiers on their dataset, utilizing different n-gram languages. The study revealed superior performance from the NN classifier with 92.8% accuracy compared to 90.3% for SVM. While these studies showed promise, the high dimensionality and sparsity of text data can impede the performance of traditional machine learning algorithms.

More recent research has delved into the application of deep learning algorithms for cyberbullying detection. Iwendi et al. [16] compared several deep learning algorithms and found that Bidirectional Long Short-Term Memory (BILSTM) achieved the highest accuracy of 82.18%. Agrawal et al. [17] utilized BILSTM and Convolutional Neural Network (CNN) on various datasets of social network messages, demonstrating outstanding performance with accuracy reaching 98% on some datasets. Similarly, Banerjee et al. [18] achieved an accuracy of 93.97% on another social media dataset using CNN.

In another study by Djuric et al. [19] a comparable methodology was employed. They made use of pre-trained word embeddings to represent online comments, enabling them to capture semantic relationships among words and enhance the identification of hate speech. For the classification task, they utilized supervised models such as SVMs and neural networks.

Hosseinmardi et al. [20] stances of cyberbullying on Instagram. The study utilized attributes including keywords, n-grams, emojis, hashtags, among others, to encode comments. Subsequently, supervised classification algorithms were employed to predict whether a given comment constituted cyberbullying or not.

These studies highlight the potential of deep learning algorithms in surpassing traditional machine learning approaches for cyberbullying detection, although traditional methods can still yield robust results. Nevertheless, there remains

DOI: 10.31695/IJASRE.2024.3.3

scope for further improvement in this domain, and the diversity of models utilized in these studies underscores the need for a comprehensive comparison of algorithms. In this paper, we undertake such a comparison to identify the best-performing algorithms, subsequently combining them in a soft voting classifier. Additionally, we explore the application of Recursive Feature Elimination with Cross-Validation (RFECV) to analyze its impact on our data, further comparing the previously used algorithms. This work serves as a foundational reference for future research, enabling the comparison of algorithm performances on similar tasks with significant implications for the advancement of cyberbullying detection.

3. PROPOSED APPROACH

For our research, we utilized a dataset originally curated by Davidson et al. [11], comprising English messages from Twitter classified into three categories: offensive, hate, and normal. The CrowdFlower platform provided the dataset's labeling. As depicted in Figure 1, the initial distribution of classes reveals an over-representation of the "offensive" class, with nearly four times more messages than the "normal" class. In contrast, the "hateful" class contains only around 1000 messages. To streamline our analysis and considering that both offensive and hateful messages can be considered forms of cyberbullying, we merged them into a single class labeled "cyberbullying." This transformation is depicted in Figure 2. However, this consolidation results in significant class imbalance, posing potential issues. Classifiers might predominantly classify messages into the dominant class, leading to a lack of accuracy in our model. To address this challenge stemming from the prevalence of cyberbullying messages, we opted to balance the classes by ensuring almost an equal number of data points in each class. This balancing operation mitigates problems associated with an imbalanced distribution. Figure 3 illustrates the class distribution after taking a 20% sample of the original "cyberbullying" class, resulting in a little over 4000 messages in each class.

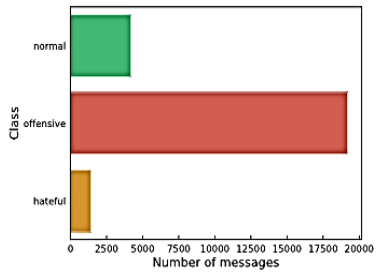


Figure 1. Initial distribution

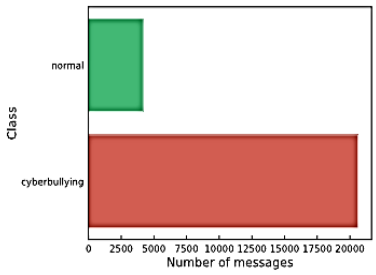


Figure 2. Distribution after the combination

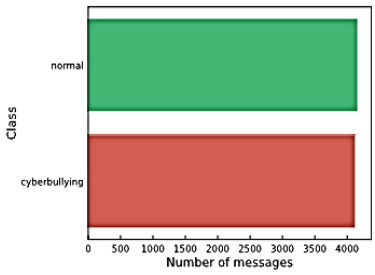


Figure 3. Distribution after resampling

3.1. Natural Language Processing (NLP)

To enhance the effectiveness of our classification, data preprocessing is essential. The initial step involves addressing contractions, such as converting "isn't" to "is not." Subsequently, we cleanse the messages by eliminating elements that contribute little to the overall meaning while maintaining message comprehensibility. This process also helps in reducing the vocabulary size of the entire dataset, leading to more accurate classification results. Specific elements removed include tags (e.g., "@" and "RT" for Twitter), http(s) and bitly links, special characters, HTML characters, and double spaces. Additionally, all characters are converted to lowercase for uniformity. However, we retain keywords located after "#" since they can carry significant meaning.

Following this step, we tokenize each word using the TweetTokenizer from the Natural Language Toolkit (NLTK) package, ensuring separate handling of each word. Next, we apply lemmatization to each word using the WordNet Lemmatizer. Lemmatization reduces each inflected word to its base form, for instance, verbs are transformed into their infinitive form, and words like "better" become "good." Concurrently, we remove stop words (e.g., "I," "you," "what," etc.) as they are exceedingly common words that add minimal semantic value to the text. This further streamlines the

DOI: 10.31695/IJASRE.2024.3.3

diversity of terms used in the dataset. Figure 4 illustrates an example of our preprocessing approach, demonstrating that the primary meaning of the message remains preserved.

Through these preprocessing steps, we ensure that the text data is appropriately cleaned and standardized, setting the foundation for more accurate and efficient cyberbullying message classification

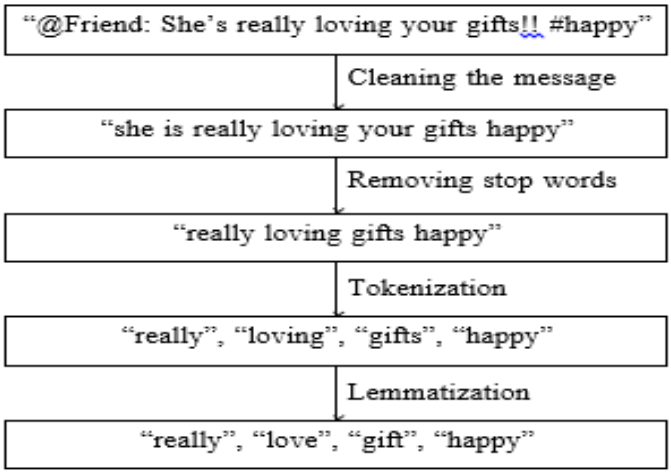


Figure 4. An Example of Message Preprocessing

3.2. Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF is a fundamental technique widely employed for feature extraction in natural language processing. It involves computing a weight for each word present in a document, which reflects its relevance concerning the other documents in the entire dataset, serving as the corpus. In our specific case, the messages themselves act as individual documents, and the entire dataset forms the corpus. The weight assigned to a particular word within a document is determined by considering two crucial factors:

1. Term Frequency (TF): This factor assesses the frequency and importance of the term's usage within the document. The formula for calculating TF, as provided in Equation (2), involves counting the occurrences of the word (Freq (i, j)) in the document and dividing it by the total number of words (L) in that document.
2. Inverse Document Frequency (IDF): The IDF factor is essential to address the significance of words that are common across multiple documents in the corpus. It reduces the weight of words that frequently appear across documents, thus giving more importance to words that appear rarely within the corpus. The IDF formula, specified in Equation (3), incorporates the total number of documents in the corpus (n) and the document frequency (dfi) of the term i, representing the count of documents in the corpus containing the term i.

By performing the multiplication of TF and IDF, as illustrated in Equation (1), we obtain the TF-IDF value, which effectively indicates the importance of a word within a specific document. This weight is then utilized as a feature representation for each message. In this representation, each term corresponds to a unique feature dimension, and its corresponding TF-IDF weight reflects the value of that feature.

$$TFIDF_{i,j} = TF_{i,j} \,.\, IDF_i \qquad \ldots\ldots\ldots \quad (1)$$

$$TF_{i,j} = \frac{\log_2(Freq(i,j)+1)}{log_2(L)} \qquad \ldots\ldots\ldots \quad (2)$$

DOI: 10.31695/IJASRE.2024.3.3

$$IDF_i = \log \frac{n}{df_i} + 1 \qquad \ldots\ldots\ldots \quad (3)$$

3.3. Deep Learning with Text Embedding

TF-IDF is a powerful technique widely used with traditional machine learning algorithms. However, when dealing with deep learning models, TF-IDF may not yield optimal performance. Deep learning models require a substantial amount of data to learn intricate patterns and relationships within the data. Using TF-IDF can result in high-dimensional and sparse feature vectors, which can negatively impact performance. To overcome this challenge, the field of Natural Language Processing (NLP) has embraced a popular technique called "deep learning with text embedding." Text embedding is a method of transforming words (represented as "w") into digital vectors "v" of a predetermined dimension. These digital vectors serve as inputs for deep learning models, particularly for tasks like text classification.

$$embedding(w) = \vec{v} \qquad \ldots\ldots\ldots \quad (4)$$

One key advantage of deep learning with embedding is its ability to capture essential semantic information present in text data. This includes capturing relationships between words and their contextual usage. The cosine similarity between two embeddings "v1" and "v2" (obtained using Equation (5)) is frequently used to measure semantic similarity between words or phrases represented as embeddings. This enables deep learning models to better understand the underlying meaning and context of the text, leading to improved performance in various NLP tasks.

$$similarity(\overrightarrow{v_1}, \overrightarrow{v_2}) = \frac{\overrightarrow{v_1}.\overrightarrow{v_2}}{\|\overrightarrow{v_1}\|\|\overrightarrow{v_2}\|} \qquad \ldots\ldots\ldots \quad (5)$$

3.4. Classification Methods

Initially, we conducted a basic classification using all the features extracted from either TF-IDF or text embedding. However, it's common for datasets to contain irrelevant or less informative features, especially in text classification tasks. Not every word contributes significantly to predicting the output variable, making feature selection a crucial step in building effective models.

In the second phase, we improved the classification process by employing the Recursive Feature Elimination with Cross-Validation (RFECV) technique. RFECV is a feature selection algorithm that assesses feature importance and eliminates less relevant features using cross-validation. The process begins by training a model with all the features and then iteratively discarding the least important ones until reaching the optimal number of features. Cross-validation is applied at each stage to estimate model performance and prevent overfitting, which can occur when using an excessive number of features. The ultimate aim is to create more efficient and interpretable models by retaining only the most meaningful features.

To carry out the feature selection in our study, we utilized two distinct algorithms. For the TF-IDF data, we employed a Logistic Regression model to identify and remove the less important features. On the other hand, for the text embedding data, we applied RFECV again, but this time using a Cat Boost model for feature selection.

3.5. Models and Metrics Used

In this research, we compared several classifiers:

- For the TF-IDF method: Logistic Regression (LR), Decision Tree (DT), Extra Trees (ET), Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naive Bayes (NB), Ada Boost (AB), Gradient Boosting (GB), Cat Boost (CB), Extreme Gradient Boosting (XGB), Light Gradient Boosting (LGB), and Soft Voting (SV). These classifiers were assessed for their performance on the task at hand to determine which model offers the best predictive capabilities based on the TF-IDF features.
- For the text embedding method, we used Convolutional Neural Network (CNN) and Bidirectional Long-Short Term Memory (BiLSTM).

DOI: 10.31695/IJASRE.2024.3.3

To assess the performance of these models, we used a comprehensive set of evaluation metrics, which included:

- Accuracy: Measures the overall correctness of the model's predictions. [21]

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \ldots\ldots\ldots \quad (6)$$

- Precision: Evaluates the proportion of true positive predictions among all positive predictions, indicating the model's ability to avoid false positives. [21]

$$Precision = \frac{TP}{TP + FP} \quad \ldots\ldots\ldots \quad (7)$$

- Recall: Also known as sensitivity, measures the proportion of true positive predictions among all actual positive instances, indicating the model's ability to capture positive cases. [21]

$$Recall = \frac{TP}{TP + FN} \quad \ldots\ldots\ldots \quad (8)$$

- F1-score: Represents the harmonic mean of precision and recall, providing a balanced measure of the model's performance. [21]

$$F1 = 2 \,.\, \frac{Precision \,.\, Recall}{Precision + Recall} \quad \ldots\ldots\ldots \quad (9)$$

- Receiver Operating Characteristic (ROC) Curve: Illustrates the model's performance across various classification thresholds, plotting the true positive rate against the false positive rate. [21]

$$TPR = \frac{TP}{P} = \frac{TP}{(TP + FN)} \quad \ldots\ldots\ldots \quad (10)$$

$$FPR = \frac{FP}{N} = \frac{FP}{(FP + TN)} \quad \ldots\ldots\ldots \quad (11)$$

By utilizing these diverse evaluation metrics, we aimed to comprehensively assess and compare the performances of the CNN and BiLSTM models in handling text embedding data.

4. EXPERIMENTAL RESULTS

We conducted our experimentation on a resampled dataset, with 20% of the data used for validation and 80% for training.

4.1. Regular Classification

In this phase of the study, we conducted experiments on a resampled dataset, allocating 20% of the data for validation and 80% for training purposes. For the TF-IDF approach, we began by tuning the hyper parameters of K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) using a grid search to identify the best settings. Afterward, we compared all the models previously mentioned.

From Figures 5a, 5b, and 5c, it is evident that several algorithms achieved high accuracy scores, around 95%. Only KNN and Naive Bayes (NB) demonstrated relatively poor performance, with accuracy scores of 84.3% and 63%, respectively. To combine the top-performing models, we utilized a soft voting classifier with the three best algorithms, namely SVM, Extreme Gradient Boosting (XGB), and Light Gradient Boosting (LGB), all having an accuracy of at least 96.2%. The soft voting technique combines the predictions from multiple models and assigns weights to each model.

DOI: 10.31695/IJASRE.2024.3.3

$$\hat{y} = arg \max_{i \epsilon 1,\ldots,k} \sum_{j=1}^{m} w_j p_{ij} \qquad \ldots\ldots\ldots \quad (12)$$

In this study, we set the weights equally for all models. The class with the highest sum of weighted probabilities is selected as the final prediction. Our implementation of this method achieved an outstanding accuracy of 96.5%, surpassing the individual performance of all other models. Moreover, the precision, recall, and F1-score at 96.5% were superior to those of the other models.

For text embedding, we utilized Convolutional Neural Network (CNN) and Bidirectional Long-Short Term Memory (BiLSTM) models. As shown in Figure 5d, both models performed well, with BiLSTM slightly outperforming CNN, achieving scores of around 96% for all evaluation metrics, while CNN had scores around 94.4%.

In Figure 5e, a general comparison of the models revealed that most of them had similar ROC curves, with an Area Under the Curve (AUC) greater than 0.98. However, as previously mentioned, NB and KNN showed inferior performance compared to other models. The soft voting classifier achieved the highest AUC value of 0.988.

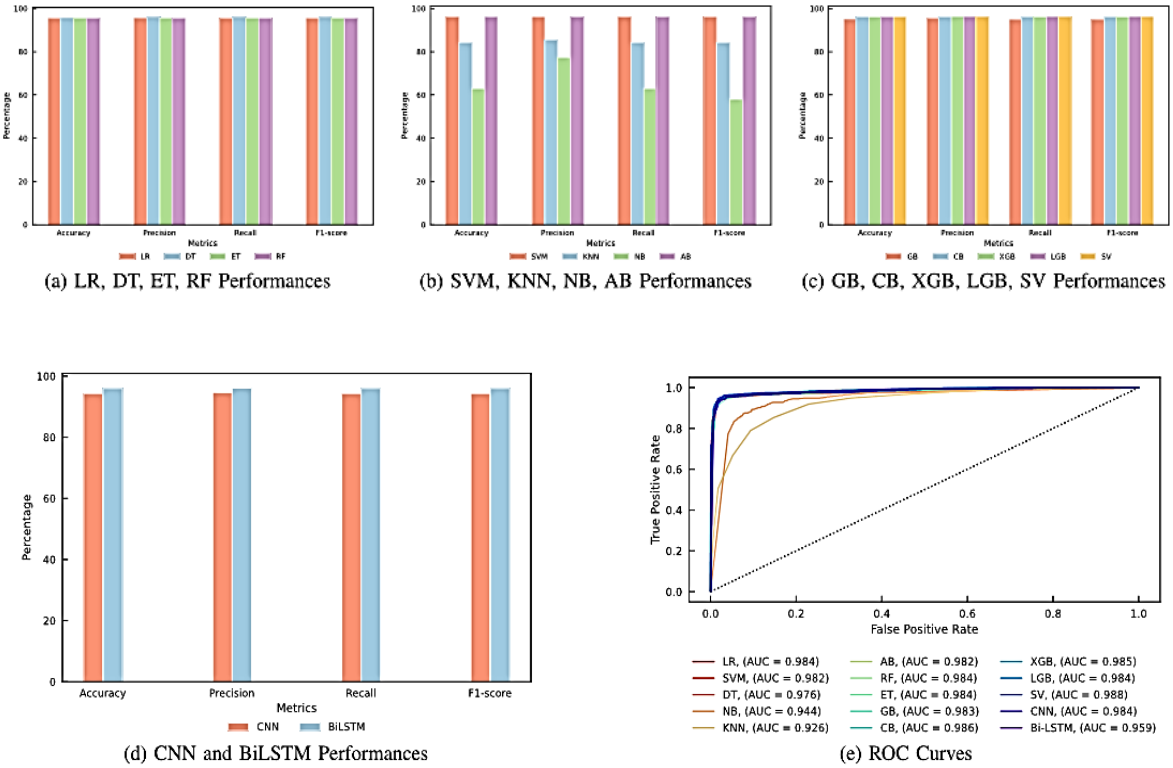


Figure 5. Models Performances without RFECV

4.2. Classification with RFECV

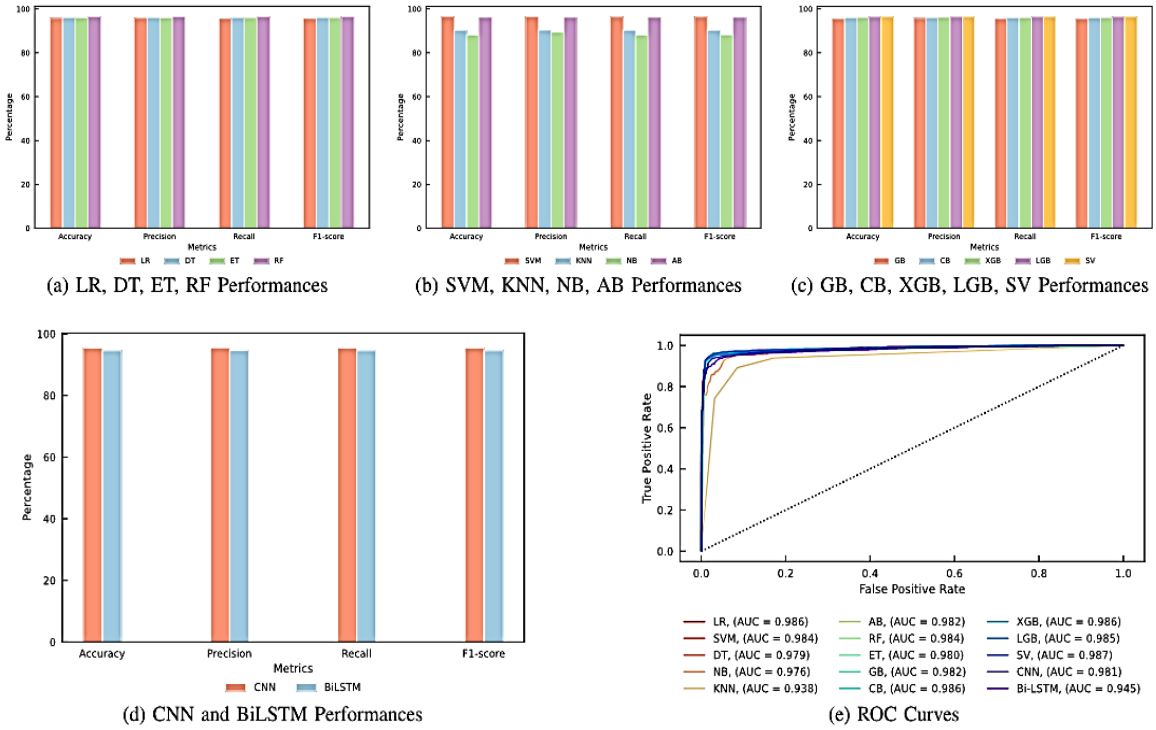
In this part of the study, we incorporated Recursive Feature Elimination with Cross-Validation (RFECV) to select an optimal subset of features, iteratively removing less important ones until obtaining the best possible feature set. For TF-IDF data, we employed a Logistic Regression (LR) model to eliminate features, reducing the vocabulary from over 9000 words to only 253. As for text embedding data, the RFECV process reduced the digital vectors from 50 to 10 features. After RFECV, we continued with the same methods used in the previous experiment.

Notably, KNN and NB demonstrated a significant increase in their metrics after RFECV, as shown in Figure 6b. NB improved by around 25% for all metrics, while KNN saw an improvement of around 6%. Despite the enhancements, their ROC curves in Figure 6e suggested that they were still trailing behind other models.

DOI: 10.31695/IJASRE.2024.3.3

For the other models, we observed minor fluctuations in their scores, as shown in Figures 6a, 6b, 6c, and 6d. Similar to the previous experiment, most ROC curves in Figure 6e had an AUC of around 0.98. We selected five models for the soft voting classifier, as their scores were now quite close to each other. The chosen models were SVM, Ada Boost (AB), Extra Trees (ET), XGB, and LGB. The soft voting classifier achieved the best overall results, with an accuracy, precision, recall, and F1-score of 96.6%, surpassing its performance without RFECV by 0.1%.

Regarding deep learning, both CNN and BiLSTM experienced a decrease in their scores. We obtained an accuracy of 95.4% for CNN and 94.5% for BiLSTM. This suggests that RFECV might not be as beneficial for deep learning models, and more data might be needed to achieve better results.



(a) LR, DT, ET, RF Performances (b) SVM, KNN, NB, AB Performances (c) GB, CB, XGB, LGB, SV Performances

(d) CNN and BiLSTM Performances (e) ROC Curves

Figure 6. Models Performances with RFECV

Overall, the results were quite similar to the previous experiment, with slightly improved results for the TF-IDF method with RFECV. The soft voting classifier (SV) achieved the best performance with an accuracy of 96.6%. Detailed results can be found in Table 1, where light-gray cells represent models chosen for the SV classifier.

Table 1 Detailed Results (%) without and with RFECV

	No RFECV				With RFECV			
	Acc	**Pre**	**Rec**	**F1**	**Acc**	**Pre**	**Rec**	**F1**
LR	95.5	95.7	95.5	95.5	95.7	95.8	95.7	95.7
SVM	96.2	96.2	96.2	96.2	96.3	96.4	96.3	96.3
DT	95.8	95.8	95.8	95.8	96.1	96.1	96.1	96.1
NB	63	77.4	63	58	88.1	89.4	88.1	88
KNN	84.3	85.2	84.3	84.2	90.3	90.4	90.3	90.3
AB	96.1	96.1	96.1	96.1	96.2	96.2	96.2	96.2

DOI: 10.31695/IJASRE.2024.3.3

RF	95.6	95.6	95.6	95.6	96.4	96.5	96.4	96.4
ET	95.4	95.4	95.4	95.4	95.8	95.9	95.8	95.8
GB	95.1	95.3	95.1	95.1	95.6	95.7	95.6	95.6
CB	96	96.1	96	96	96	96.1	96	96
XGB	96.2	96.3	96.2	96.2	96.1	96.2	96.1	96.1
LGB	96.3	96.3	96.3	96.3	96.3	96.3	96.3	96.3
SV	96.5	96.5	96.5	96.5	96.6	96.6	96.6	96.6
CNN	94.3	94.6	94.3	94.3	95.4	95.5	95.4	95.4
BiLSTM	96	96.1	96	96	94.5	94.6	94.5	94.5

5. CONCLUSION

In this research paper, we proposed a machine learning-based approach to detect cyberbullying-related messages in our dataset, utilizing advanced Natural Language Processing (NLP) techniques and experimenting with two feature extraction methods: TF-IDF and text embedding. To tackle class imbalance, we employed resampling techniques. Our findings revealed that the soft voting classifier, combining SVM, XGB, and LGB, achieved the best performance with an impressive accuracy of 96.5%. Additionally, the BiLSTM and CNN models using text embedding also produced high scores.

To further enhance model performance, we employed Recursive Feature Elimination with Cross-Validation (RFECV) to select the most relevant features. This led to a significant reduction in the TF-IDF vocabulary to 253 words and notably improved the performance of KNN and NB. However, we observed that deep learning models such as BiLSTM and CNN did not benefit significantly from feature reduction, as they inherently learn crucial features from the input data. Nevertheless, the soft voting classifier, when used with SVM, AB, ET, XGB, and LGB, experienced a slight improvement, achieving an accuracy of 96.6%.

Our study serves as a valuable starting point for future research in this domain, as we extensively compared various models and showcased the effectiveness of machine learning techniques in identifying cyberbullying messages within the given dataset. RFECV demonstrated its potential when utilized with traditional machine learning models. Future investigations could apply our approach to different datasets for classification tasks, and gathering additional data could further enhance the performance of deep learning models. Moreover, exploring other advanced models like BERT (Bidirectional Encoder Representations from Transformers) could be an interesting avenue to explore.

Overall, our findings contribute to the development of more robust models for similar tasks, leading to improved predictions and more informed decision-making processes.

REFERENCES

[1] L. R. Betts," Cyberbullying: Approaches, consequences, and interventions," *Springer*, 2016, https://doi.org/10.1057/978-1-137-50009-0.

[2] D. Olweus,"Cyberbullying: An overrated phenomenon?" *European Journal of Developmental Psychology*, vol. 9, no. 5, pp. 520-538, (2012), https://doi.org/10.1080/17405629.2012.682358 .

[3] C. P. Barlett, M. M. Simmers, B. Roth and D. Gentile, "Comparing cyberbullying prevalence and process before and during the COVID-19 pandemic," *The Journal of Social Psychology*, vol. 161, no. 4, pp. 408-418, 2021, https://doi.org/10.1080/00224545.2021.1918619.

[4] S. Hinduja and J. W. Patchin, "Bullying, cyberbullying, and suicide," *Archives of Suicide Research*, vol. 14, no. 3, pp. 206-221, 2010,https://cyberbullying.org/bullying-cyberbullying-and-suicide.

DOI: 10.31695/IJASRE.2024.3.3

[5] R. M. Kowalski, G. W. Giumetti, A. N. Schroeder and M. R. Lattanner, "Bullying in the digital age: A critical review and meta-analysis of cyberbullying research among youth," *Psychological Bulletin*, vol. 140, no. 4, p. 1073, 2014, http://dx.doi.org/10.1037/a0035618.

[6] S. Keith and M. E. Martin, "Cyber-bullying: Creating a culture. Reclaiming Children and Youth," *The Journal of Strength-based Interventions*, vol. 134, no. 3, pp. 224-228, 2005, https://www.proquest.com/openview/b2c7ccb1ffef391fa65b850973adbbfd/1?pq-origsite=gscholar.

[7] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell and N. Tippett, "Cyberbullying: Its nature and impact in secondary school pupils," *Journal of Child Psychology and Psychiatry*, vol. 49, no. 4, pp. 376-385, 2008,https://doi.org/10.1111/j.1469-7610.2007.01846.x.

[8] R. Slonje, P. K. Smith and A. Frisen, "The nature of cyberbullying, and strategies for prevention," *Computers in Human Behavior*, vol. 29, no. 1, pp. 26-32, 2013, https://psycnet.apa.org/doi/10.1016/j.chb.2012.05.024.

[9] R. Donegan, "Bullying and cyberbullying: History, statistics, law, prevention, and analysis," *The Elon Journal of Undergraduate Research in Communications*, vol. 3, no. 1, pp. 33-42, 2012, https://eloncdn.blob.core.windows.net/eu3/sites/153/2017/06/04DoneganEJSpring12.pdf.

[10] T. Diamanduros, E. Downs and S. J. Jenkins, "The role of school psychologists in the assessment, prevention, and intervention of cyberbullying," *Psychology in the School*s, vol. 45, no. 8, pp. 693-704, 2008, https://doi.org/10.1002/pits.20335.

[11] T. Davidson, D. Warmsley, M. Macy and I. Weber, "Automated hate speech detection and the problem of offensive language," *Proceedings of the 11th International AAAI Conference on Web and Social Media*, pp. 512-515, 2017, https://doi.org/10.48550/arXiv.1703.04009.

[12] M. M. Islam, M. A. Uddin, L. Islam, A. Akter, S. Sharmin and U. K. Acharjee, "Cyberbullying detection on social networks using machine learning approaches," *In 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE). IEEE*, pp. 1-6, 2020,http://dx.doi.org/10.1109/CSDE50874.2020.9411601.

[13] A. Muneer and S. M. Fati, "A comparative analysis of machine learning techniques for cyberbullying detection on Twitter," *Future Internet*, vol. 12, no. 11, p. 187, 2020, http://dx.doi.org/10.3390/fi12110187.

[14] B. S. Nandhini, and J. Sheeba, "Cyberbullying detection and classification using information retrieval algorithm," *In Proceedings of the 2015 international conference on advanced research in computer science engineering & technology (ICARCSET 2015)*, pp. 1-5, 2015,https://doi.org/10.1145/2743065.2743085.

[15] J. Hani, M. Nashaat, M. Ahmed, Z. Emad ,E. Amer and A. Mohammed, "Social media cyberbullying detection using machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5. 2019, http://dx.doi.org/10.14569/IJACSA.2019.0100587.

[16] C. Iwendi, G. Srivastava, S. Khan and P. K. R. Maddikunta, "Cyberbullying detection solutions based on deep learning architectures," *Multimedia Systems*, vol 29 ,Iss 3,pp 1839–1852, 2020, https://doi.org/10.1007/s00530-020-00701-5.

[17] S. Agrawal and A. Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms," *In Advances in Information Retrieval: 40th European Conference on IR Research, ECIR 2018*, Grenoble, France, March 26-29, 2018, Proceedings. Springer, pp. 141-153, https://link.springer.com/chapter/10.1007/978-3-319-76941-7_11 .

[18] V. Banerjee, J. Telavane, P. Gaikwad and P. Vartak, " Detection of cyberbullying using deep neural network," *In 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS). IEEE*, pp. 604-607, 2019, https://ieeexplore.ieee.org/document/8728378.

DOI: 10.31695/IJASRE.2024.3.3

[19] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic and N. Bhamidipati, "Hate Speech Detection with Comment Embeddings," *In WWW '15 Companion: Proceedings of the 24th International Conference on World Wide Web*, pages 29-30, 2015, https://dl.acm.org/doi/10.1145/2740908.2742760.

[20] H. Hosseinmardi, S. Arredondo Mattson, R. I. Rafiq, R. Han, Q. Lv and S. Mishra, "Detecting cyberbullying incidents on the Instagram social network," *In Proceedings of the 11th International Conference on Web and Social Media. Association for Computational Linguistics*, 2015, https://www.researchgate.net/publication/273640275.

[21] OpenAI. ChatGPT (Version GPT-3.5), 2021, Retrieved from https://openai.com/

E-mail: jinan_redha@uomustansiriyah.edu.iq

DOI: 10.31695/IJASRE.2024.3.3