machine learning & knowledge extraction

Article

An Ensemble-Based Multi-Classification Machine Learning Classifiers Approach to Detect Multiple Classes of Cyberbullying

Abdulkarim Faraj Alqahtani [1,2,*] and Mohammad Ilyas [1,*]

[1] Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431, USA

[2] Ministry of National Guard, King Khalid Military Academy, Riyadh 14625, Saudi Arabia

* Correspondence: aalqahtani2021@fau.edu (A.F.A.); ilyas@fau.edu (M.I.)

Abstract: The impact of communication through social media is currently considered a significant social issue. This issue can lead to inappropriate behavior using social media, which is referred to as cyberbullying. Automated systems are capable of efficiently identifying cyberbullying and performing sentiment analysis on social media platforms. This study focuses on enhancing a system to detect six types of cyberbullying tweets. Employing multi-classification algorithms on a cyberbullying dataset, our approach achieved high accuracy, particularly with the TF-IDF (bigram) feature extraction. Our experiment achieved high performance compared with that stated for previous experiments on the same dataset. Two ensemble machine learning methods, employing the N-gram with TF-IDF feature-extraction technique, demonstrated superior performance in classification. Three popular multi-classification algorithms: Decision Trees, Random Forest, and XGBoost, were combined into two varied ensemble methods separately. These ensemble classifiers demonstrated superior performance compared to traditional machine learning classifier models. The stacking classifier reached 90.71% accuracy and the voting classifier 90.44%. The results of the experiments showed that the framework can detect six different types of cyberbullying more efficiently, with an accuracy rate of 0.9071.

Keywords: ensemble models; cyberbullying; multi-classification; multiclass; TF-IDF; N-gram; stacking classifier; voting classifier

check for updates

Citation: Alqahtani, A.F.; Ilyas, M. An Ensemble-Based Multi-Classification Machine Learning Classifiers Approach to Detect Multiple Classes of Cyberbullying. *Mach. Learn. Knowl. Extr.* **2024**, *6*, 156–170. https://doi.org/10.3390/make6010009

Academic Editor: Simon Tjoa

Received: 26 October 2023
Revised: 31 December 2023
Accepted: 9 January 2024
Published: 12 January 2024

1. Introduction

Currently, social platforms are prevalent in most people's lives, as they allow them to express and comment on their views. Moreover, most people spend a long time on these platforms and use them to communicate. Online platforms allow users to share their comments, blogs, images, and videos publicly for viewing. While there are many benefits to this communication, it can also have harmful aspects, such as cyberbullying, which has increased as people spend more time on social platforms. Cyberbullying is a behavior that results in social attacks, and it is a high source of risk, which is generated through these social platforms. In addition, with the massive increase in users on online platforms, the volume of data has also surged. In 2018, Twitter alone had around 330 million active users, generating 550 million tweets daily. While these tweets offer diverse research opportunities, they can also contain offensive content, criminal activities, and instances of bullying [1].

According to the authors in [2], one reason for the increase in cyberbullying is that people spend much time on online platforms, which leads them to hurt each other when reaching out through these platforms. In addition, the authors emphasized that not all the content that is posted and viewed is appropriate for all people, so they reply with tweets that contain hurtful words because of their conflicting viewpoints or even only to simply engage in the act of bullying. Thus, using machine learning (ML) to detect cyberbullying on online platforms has become necessary to avoid these unwanted behaviors. However, the detection of cyberbullying faces challenges such as the massive amount of data, and some data need to be detected using binary classification, while others need to be detected using multi-classification, depending on the number of target classes. Additionally, data categories are subject to change, implying that each type of data should be paired with a suitable model to yield effective detection performance. For instance, if the data are categorized into binary labels, binary models optimized for such data are essential. The same principle applies to data classified into multiple categories [3]. Moreover, the authors in [4] mentioned that the complexity of detection increases when dealing with datasets containing multiple categories, as the classification process tends to be time-consuming. Also, they stated that most existing experiments on detecting cyberbullying are specific, such as models classifying tweets as bullying or non-bullying. Types of cyberbullying have been allocated into many categories, such as age, religion, race, gender, and ethnicity. Thus, multi-classification techniques are expected to improve the detection of these types of cyberbullying.

The authors in [5] indicated that cyberbullying has become prevalent, and the detection of cyberbullying has become complex. This is because there are many tweets that contain various kinds of cyberbullying depending on the context of these tweets, which may indicate gender, religion, age, ethnicity, or other types of cyberbullying. Training models to detect specific types of cyberbullying require a different approach than training models to categorize tweets as either free of cyberbullying (0) or containing cyberbullying (1) based on certain words, making it challenging to identify the specific types of cyberbullying.

In addition, some of the challenges that have been discussed when using multiple classes to analyze text data are related to our work in this paper as we are dealing with tweets. The representation of text is different depending on the field in which the text is expressed. This means that, in the preprocessing stage, the feature extraction will be various, so the bag of words feature may achieve optimal performance when analyzing text related to customer reviews or other types of text; however, this feature may not be beneficial when preprocessing cyberbullying text. Thus, when training models to detect multiclass datasets, they need to test several feature extractions to find suitable features that work with the concept of text [6].

Another challenge that needs to be mentioned when dealing with multiclass datasets is finding a need for the correction of tweets that were written. Extracting the context of tweets or text is complex and requires a high level of training. The authors in [7] suggested using manual effort to deal with text from experts in this field, which involves understanding the meaning of the writers and training models to understand upcoming tweets with the same concept.

Cyberbullying is difficult and complex to detect manually by humans; thus, an automatic system can help detect text that contains offensive phrases through ML algorithms. Building systems to detect cyberbullying is not a new experiment or field of study, as many frameworks and methodologies have been suggested to solve this issue. However, the increase in the amount of data and cyberbullying that have appeared in recent years has inspired collaboration to improve the performance of models that can detect cyberbullying. As mentioned earlier, there are multiple types of cyberbullying datasets labeled as multiclass cyberbullying, which is more challenging to detect compared to datasets labeled as binary.

In this paper, we contribute to the development of ensemble models by employing three multi-classification models to detect multiclass cyberbullying in a dataset. Firstly, we tested three multi-classification models—RF, DT, and XGBoost—utilizing the TF-IDF feature-extraction method, and combined these models into two ensemble techniques. Secondly, we aimed to explore and investigate the most-suitable ML techniques for dealing with multiclass cyberbullying datasets compared to traditional ML models. Thirdly, we employed N-gram feature engineering with TF-IDF, achieving state-of-the-art results, particularly with bigrams. Fourthly, our proposed model achieved better performance compared to prior experiments that utilized the same dataset. Based on our investigation, this new framework combines multi-classification models to detect multiclass cyberbullying in a dataset and demonstrates satisfactory performance.

2. Literature Review

Many experiments have contributed to this field using various methodologies and frameworks. This section will summarize some previous research work closely related to cyberbullying detection on online platforms. The authors in [8] developed bagging ensemble models that achieved satisfactory accuracy while classifying binary cyberbullying datasets. The highest accuracy achieved in their experiment was 96 % by using TF-IDF, which is one of the feature-extraction techniques that uses words in documents. They used this feature with unigrams, which deal with each word in the document as one token. The authors emphasized that their aim in this paper was to develop a voting model that involves double and single ensemble-based to detect the content of text and classify it as either 'offensive' or 'non-offensive'. They mentioned in their paper that the dataset classified in their experiment was collected from the Twitter platform and contains 9093 tweets. They completed many preprocessing steps to make it easy for the models to classify the architecture of the model, which uses an ensemble model that combines seven classifiers in different methods, namely a single-level ensemble model that involves all seven classifiers in one novel model. In addition, the double-level ensemble model involves four ML algorithms in one novel model, and the other three models are combined in a second novel model; finally, these two novel models are integrated into one novel model.

Moreover, while discussing prior research work related to detecting cyberbullying, the authors in [9] experimented with seven ML algorithms to detect cyberbullying tweets in a Twitter dataset containing 37,373 unique tweets. Based on their experiment, the highest accuracy achieved was around 90.57 %, which resulted from linear regression. Also, they applied two feature extractions addressed in their experiment, TF-IDF and Word2Vec, to enhance the classifier's performance. The model approaches in their experiment start by importing the dataset and executing the preprocessing phase, including removing stop words, punctuation, special characters, and stemming. After the preprocessing steps, they split the dataset into training and testing sets and applied TF-IDF for the training set and Word2Vec for the testing set. The last step was running the models to test the prediction and evolution of the model. The reason for using two different feature extractions was to examine which feature performed well in their classification. They stated that TF-IDF is preferable with large data while Word2Vec can excel with small data, as, in general, testing sets mainly involve small data. They specified a limitation in their study, which is real-time detection, while they were able to investigate an appropriate feature-extraction method that performed well in classification.

Many techniques have been experimented with to detect cyberbullying in online platforms. Ahamed et al. [10] developed an automatic system that detects cyberbullying tweets using a voting ensemble model that combines three classifier networks, the well-known RoBERTa, XLNet, and GPT2, to detect multiclass datasets. The authors emphasized that using an ensemble model was the best choice in their experiment. The ensemble approaches used in their experiment are called hard voting, which is voting based on similarity, and soft voting, which is voting based on averaging. They used two different datasets related to cyberbullying on the Twitter platform and tested their model on these datasets separately. The first dataset was unbalanced, and the second one was balanced to evaluate their model. The result achieved on the first dataset by the ensemble model was 85.81 % for detecting six classes related to cyberbullying, and the accuracy of the second dataset was 87.48 % for detecting three classes.

In another research paper that used the same dataset we used, the authors in [11] developed a deep analysis approach to detect six types of cyberbullying. They utilized a variety of five machine learning algorithms using text-based feature extraction. Based on their experimental results, LightGBM demonstrated the best performance in a range between 84.49% and 85.5% for the accuracy, precision, recall, and F-1 score. This study aligns with our experiment in utilizing machine learning algorithms, as we did, and including all classes of the dataset, mirroring our approach. The feature extractions are reasonable for improving the performance of the model. The authors in [12] tried various types of feature extractions to achieve the best accuracy when testing models that detect cyberbullying texts in their experiment. The first feature is applying the TF-IDF feature to make the texts as suitably represented data when feeding the model, which is a practical feature for converting text into numerical data. The second feature uses the sentiment analysis method to extract the polarity of the text or sentences, and the last feature is N-grams, which are sequences of words that will be added as one token depending on the value of n. TF-IDF's primary concept is to convert the text to numerical data and determine the relative importance of individual words within a given passage. Also, they applied the sentiment analysis technique to determine the polarity of the sentences, which they then included as a feature alongside TF-IDF's existing characteristics. To determine whether a sentence should be labeled as positive or negative, they used a polarity function with the Textblob library, which is a pre-trained model for extracting the polarity in textual data. In addition, they used TF-IDF to obtain the features and sentiment polarity. They emphasized their proposed approach of using N-grams to look at the different ways words can be put together when evaluating the model. The result achieved in their experiment by Support Vector Machine (SVM) when using 4-grams was 90.3 % and by using a Neural Network (NN) when using 3-grams, it was 92.8 %.

Capturing the meaning of text poses a significant challenge in detecting cyberbullying. The research study [13] addressed various challenges in cyberbullying detection, with a particular focus on understanding the meaning of context. They presented a compilation of previous works with a focus on improving contextual understanding, including several models and word representations that have contributed to resolving this issue. Detecting cyberbullying is a complex task that requires capturing context. According to this research study, word embedding proved to be an effective tool in conjunction with deep learning, while TF-IDF demonstrated strong performance when applied in machine learning scenarios. While the above prior works suggested to develop automatic systems by ML or deep learning to show the outputs for detecting cyberbullying, our experiment suggests improvements in the accuracy and uses feature extraction. This study contributes to the review of previous experiments that need to optimize their accuracy and efficiency for detecting bullying tweets, whether multiclass or binary class, using feature-extraction methods that enhance the performance. In addition, this study focused on combining several supervised ML classifiers for multi-classification using features that help to enhance the classification. Table 1 shows our experiment compared with the others.

Table 1. Comparison of our proposed approach with other methods using the same dataset.

Citation	Year	Models	Accuracy	Precision	Recall	F1-Score
[10]	2022	Max-voting ensemble	85.25%	85.02%	85.25%	85.10%
[11]	2022	LightGBM	85.05%	84.00%	85.00%	84.49%
Our proposed models	2023	Voting Stacking	90.41% 90.71%	90.69% 90.08%	90.36% 90.60%	90.45% 90.63%

Table 1 illustrates the results of our proposed approach compared to the recent approaches in [10,11] concerning four evaluation metrics using the same dataset. The table gives a thorough summary of each model's performance by summarizing their accuracy, precision, recall, and F1-Score. We applied k-fold cross-validation to assess the efficiency of various configurations and ensure the optimal performance of our approach. Specifically, we set k = 10 for the ensemble models, providing a robust evaluation across multiple folds [14]. Hence, in comparison to the alternative experiments, our proposed models outperformed them consistently in the conducted experiments.

Background Ensemble Models for Detection of Text

Ensemble models have been used in many fields for detection using ML. The authors in [15] developed an ensemble model to detect cyberbullying in their new dataset with binary classification as their dataset was categorized using binary labels. In the first step, they attempted to improve upon SVM. For their second strategy, they relied on DistilBERT, a more-efficient and -compact rendition of the transformer model BERT. When they combined the first three models, they obtained two more ensemble models. In comparison with the other three models, they found that the ensemble models outperformed the base model on all evaluation metrics except precision. With an ensemble model, they achieved an accuracy of 89.6%, surpassing the SVM model, which yielded an accuracy of 85.53%. The DistilBERT model achieved the highest accuracy of 91.17%. In their experiment, various TF-IDF feature-extraction approaches, including word, character, and N-gram sequencing, were empirically evaluated and compared using an SVM model. In another example of ensemble models for analyzing detection, the authors in [16] proposed a stacking model to detect short message service (SMS) spam. They identified stacking as a method of ensemble learning used to improve model predictions by aggregating the results of numerous models and passing them through another ML model. They developed an ensemble model with bag of words feature extraction to enhance their model's performance, naming it AstNB—a new augmentation and stacking approach combined with the transfer learning approach of Naive Bayes (NB). Their goal in this experiment was to detect SMS spam across multiple datasets, achieving the highest accuracy of 98.1% for detecting spam SMS domains.

3. Methodology

3.1. Dataset Description

In our experiment, while various datasets contain cyberbullying text, we proposed using recent cyberbullying instances to identify new textual concepts and the idioms that most strongly indicate cyberbullying. Additionally, most prior research focused on binary datasets, which are less complex compared to multiclass datasets for text analysis, as mentioned earlier. Moreover, given the diverse types of cyberbullying, addressing this issue has become more challenging. Therefore, our aim was to utilize a recently released multiclass dataset, publicly available on Kaggle, collected by the authors [17]. The dataset was generated from the Twitter platform before the name changed to X, which is an appropriate choice for analyzing a variety of tweets and a universal social media platform that involves diverse users from different regions with diverse cultures, religions, genders, and ages. About 48,000 tweets were used to compile the dataset, which was labeled into six classes based on its fields: age; gender; ethnicity; religion; other, which indicates other types of cyberbullying; and not_cyberbullying, indicating the absence of cyberbullying. Figure 1 illustrates the distributions of these classes, and Table 2 presents one actual sample of each type of tweet. For privacy and offensive content concerns, some letters in the samples have been replaced with asterisks (*). The dataset involves two columns: one for tweets and the other for labels representing the six types of cyberbullying. Table 2 provides a real sample of tweets from the dataset.

3.2. Supervised Machine Learning Models

Three commonly used supervised machine learning classifiers for multi-classification were chosen to assess the classification performance in our experiment. This section will provide a brief description of each of these classifiers.

3.2.1. Decision Tree

DT is a type of regression tree model. It gradually develops a Decision Tree in tandem with a subgroup of a dataset into ever-tinier pieces [18]. The result is a tree structure with both decision nodes and leaf nodes. In addition, DT is the most-popular classifier model, which can predict and detect problems; it is also considered powerful and easy to use, and decisions can be made quickly and easily based on the data [19].

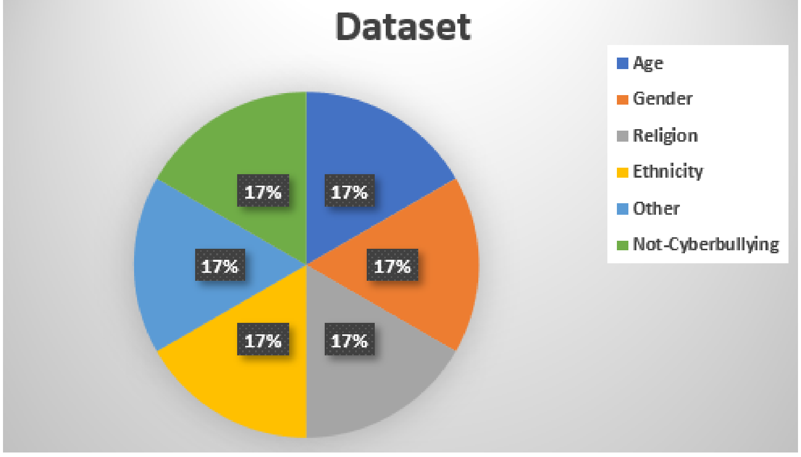

Figure 1. The distribution of the dataset.

Table 2. Sample tweets from the dataset.

Label	Tweet
Age	these are the girls who bullied me in high school.
Gender	I'll still call them females. And B ****** too.
Religion	Yes, unlike the gulf Muslim countries where they still beh *** and won't let women drive.
Ethnicity	What about Asian Americans? Anti asian racism is on the raise and you have done nothing. Are we not colored enough for u?.
Other	I realized he gets bullied... that's just more of a reason for him to be my friend.
Not Cyberbullying	This has been the longest, most uneventful weekend of my life. I feel like I just came from a vacation break.

3.2.2. Random Forest

RF combines multiple Decision Trees from diverse datasets to boost classification accuracy [19]. It operates as an ensemble, wherein individual Decision Trees collaborate as a group. Employing a bagging approach, the algorithm aggregates learning models to enhance the overall results. RF is known for its simplicity in construction and formulation [20].

3.2.3. XGBoost

The XGBoost classifier extends the Gradient Boosting Decision Tree (GBDT) model for enhanced performance. By combining multiple Decision Trees, XGBoost improves accuracy. This algorithm employs distributed gradient boosting and is designed to be fast, flexible, and user friendly [21].

3.2.4. Voting Classifier

The VC is a model that learns from an ensemble of other models and, then, makes a prediction about the output class by selecting the class with the greatest probability [22]. The VC combines the features that help predict the class based on the probabilities extracted from these features, and the classifier decides to predict the output of the class based on the highest probability that has been voted on [23]. The VC supports two types: hard voting, which relies on the highest majority, and soft voting, which considers the average probability. In this study, we employed hard voting to combine DT, RF, and XGBoost for classification.

3.2.5. Stacking Classifier

The SC uses meta-classifier strategies to integrate various classification models. Also, the SC involves combining the results of many estimators and, then, using a classifier to make a single prediction. Stacking is choosing the best features of many estimators at once by feeding their combined output into an individual classification model [23].

3.3. Ensemble Learning

The authors in [24] emphasized that ensemble learning (EL) significantly contributed to the improved performance of their experiment. They detailed an EL approach that combines four supervised ML algorithms: Decision Tree, Random Forest, Logistic Regression, and K-Nearest Neighbor, employing two ensemble techniques. Their approach involves combining these classifiers into a unified ensemble method. Ensemble Learning is a technique that involves grouping multiple ML algorithms to enhance model evaluation accuracy. In our experiment, we combined three popular supervised ML algorithms tailored for multiclass target labels. Figure 2 illustrates the workflow of our ensemble approach.

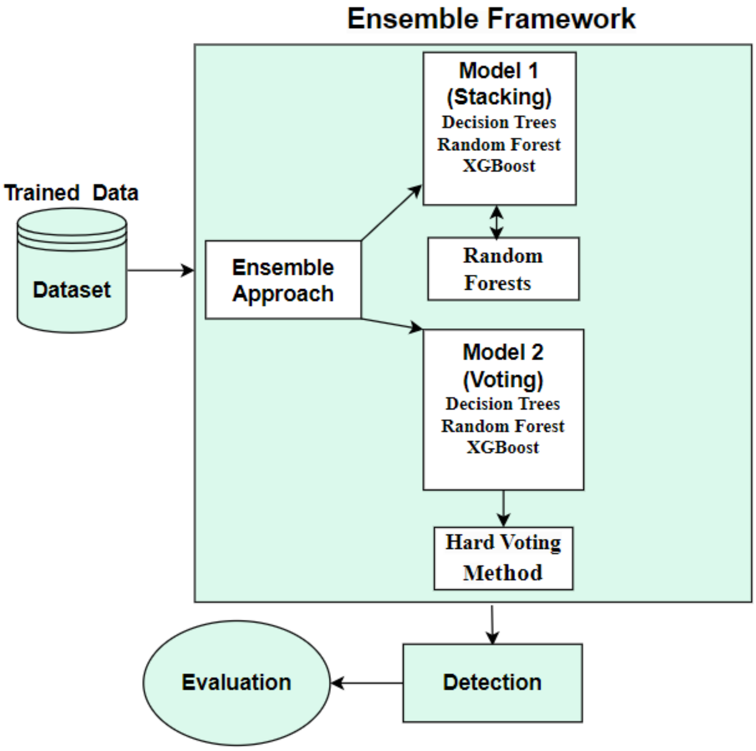

Figure 2. The workflow of our ensemble framework.

3.4. Data Preprocessing

The success of an ML model heavily relies on the quality of the training data, underscoring the significance of data preprocessing in the development process. Effective data cleaning is essential to ensure clarity and prevent accuracy degradation when inputting data into the model. Leveraging the NLTK library, a widely used tool for data preprocessing, we meticulously cleaned the data. Through the NLTK, we eliminated undesirable elements such as tags, hashtags, links, duplicates, punctuation, and numbers. Furthermore, all tweets were uniformly converted to lowercase for consistency. Furthermore, maintaining an equal number of classes is crucial for achieving satisfactory system performance with TF-IDF feature extraction, as it is optimized when the classes are of the same quantity [25]. After completing the dataset cleaning process, we partitioned the data into a test set (20%) and a training set (80%). We experimented with various test set percentages, including 10%, 30%, and 40%, and found that the 20% test set yielded the highest accuracy. The final preprocessing step involved applying feature extraction, which will be discussed in detail in the following section. Figure 3 illustrates the workflow of our classification methodology, while Figure 4 provides an overview of the preprocessing steps applied to all tweets in the dataset.

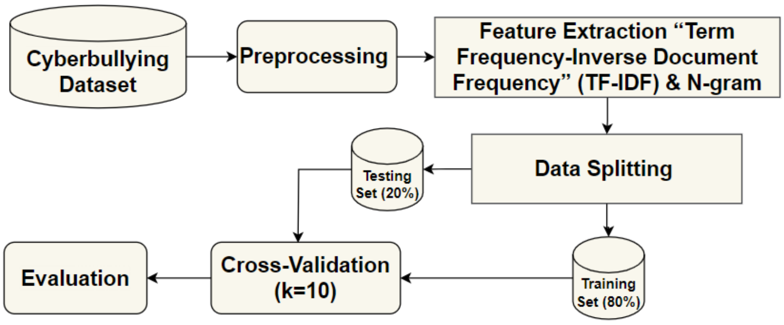

Figure 3. The workflow of the methodology of cyberbullying classification.

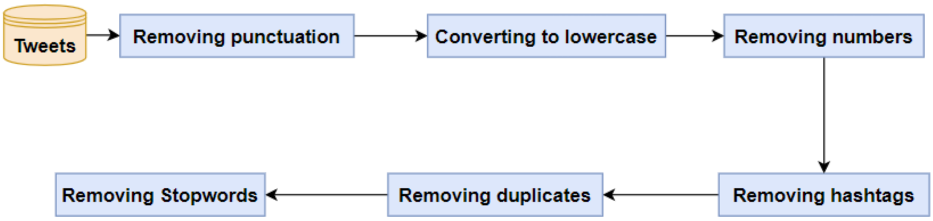

Figure 4. The preprocessing steps for tweets of the cyberbullying dataset.

3.5. Feature Extraction

The first phase involves preparing text data for input into ML algorithms. We utilized TF-IDF [26] to extract features and store the data in a feature list. TF-IDF analyzes the text, determining the relative importance of words and sentences in tweets, and identifies the most-frequent words in the document. Additionally, we incorporate the N-gram feature with TF-IDF, a common method for consistently representing texts and capturing stylistic aspects of the text concept [27]. The authors in [28] utilized Word TF-IDF, N-gram TF-IDF, and Char TF-IDF as instances of TF-IDF vectors constructed with different levels of input tokens. In our experiment, we explored three N-gram types: unigram, bigram, and trigram. Notably, the highest accuracy was achieved when using bigrams. We opted for TF-IDF due to its efficiency in handling vocabulary and better management of the word frequency in text [29]. While there are alternative methods like Word2Vec, developed by Mikolov [30], this embedding may not be optimal for certain vocabularies in text as it has been exclusively trained on Wikipedia vocabulary [31,32]. Moreover, several experiments have emphasized that TF-IDF consistently outperforms other feature-extraction methods. In [33], the authors noted that TF-IDF achieved the highest accuracy in their approach compared to bag of words (BoW) feature extraction.

Term Frequency Inverse Document Frequency

TF-IDF is a method of feature engineering used to extract features from text data, thereby enhancing context. This method is widely popular in the field of text analysis. In TF-IDF, each term in the document is assigned a numerical representation, determined by a weight based on both term frequency (TF) and inverse document frequency (IDF) features. Words with higher weights carry more significance in the document compared to those with lower weights [34]. To calculate TF-IDF, TF and IDF must be obtained separately. Term frequency (TF) is often used to determine the weight of a term. Equation (1) shows how TF is calculated. Equation (1):

$$\mathrm{TF}_{t,d} = \frac{n_{t,d}}{\sum_k n_{t,d}} \tag{1}$$

Term Frequency ($\text{TF}_{t,d}$) is determined by spreading out the total number of a specific term t in a document d ($n_{t,d}$) by the total number of terms in the document ($\sum_k n_{t,d}$). This equation quantifies the frequency of the term relative to the overall term count in the document.

The above equation elucidates the process of determining TF, where each term in the document is assigned a numerical representation. In contrast, IDF calculates the representation value for terms that are uncommon in the corpus, as referenced in [19]. This implies that, when rare or uncommon words appear in one or more documents, these words carry significant and meaningful information. Equation (2) outlines the calculation of the IDF weight. Equation (2):

$$\text{IDF}(t) = \log\left(\frac{N}{d_{ft}}\right) \tag{2}$$

Inverse Document Frequency (IDF) is calculated to identify the score of the total number of documents (N) and (df_t), which is represented by the number of documents containing the term.

In the end, we just multiply the TF by the IDF in the following way to obtain the TF-IDF weight of the phrase (t) for every term in the corpus, as shown in Equation (3):

$$\text{TF-IDF} = \text{TF} \times \text{IDF} \tag{3}$$

Now, let us consider the calculation of the TF-IDF value for the term 'Hate' using the formula. Assume that 'Hate' appears 4 times in a document containing 21 words. In this case, we first need to calculate the term frequency (TF) for the word 'Hate', which would be TF = $\frac{4}{21}$ = 0.19. IDF measures the significance of terms based on their rarity across all documents [35]. Thus, If there are 250 documents in total and 150 of them include the term 'Hate', the IDF value for the word 'Hate' would be $\text{IDF} = \log\left(\frac{250}{150}\right) = 0.22$. Finally, we need to obtain the TF-IDF value by multiplying the TF by the IDF, so that the TF-IDF score for the term 'Hate' would be represented as $\text{TF-IDF} = 0.19 \times 0.22 = 0.04$

3.6. Proposed Framework

We propose two ensemble techniques that involve three multi-classification models to detect multiclass cyberbullying tweets. In our framework, we explored various feature-extraction methods to achieve optimal performance, such as the N-gram feature with TF-IDF. Following feature extraction, the obtained features are input into a classification algorithm for training and testing before the classifier is utilized in the prediction phase of the proposed method. Our framework employs an ensemble method to achieve performance results from three multi-classification classifiers combined into two different types of ensemble techniques: voting and stacking. The multi-classification classifiers are Decision Tree, Random Forests, and XGBoost, combined into one ensemble classifier. Figure 5 illustrates the workflow of the methodology of our proposed ensemble approach.

In our proposed approach, we applied hard voting when using an ensemble voting classifier to classify a multiclass cyberbullying dataset. Hard voting involves using the majority to classify a class during prediction. With the dataset containing six classes, each class is classified based on the majority of votes, leading to improved accuracy. Additionally, when employing the ensemble as stacking, we designated the Random Forest (RF) classifier as the base for the stacking technique because it achieved the highest accuracy. The stacking process involves taking predictions made by base classifiers and combining them to create a new set of features, and we chose RF as the base for this ensemble technique.

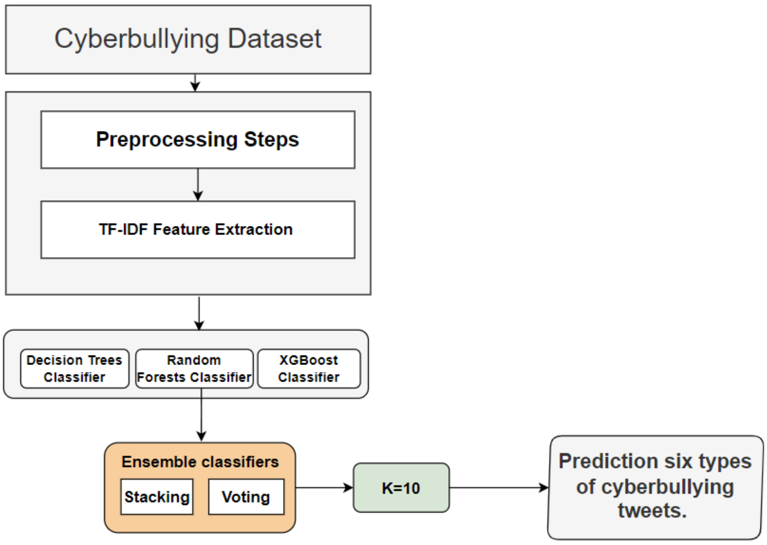
Cyberbullying Dataset
Preprocessing Steps
TF-IDF Feature Extraction
Decision Trees Classifier
Random Forests Classifier
XGBoost Classifier
Ensemble classifiers
Stacking
Voting
K=10
Prediction six types of cyberbullying tweets.

Figure 5. Our proposed ensemble approach.

4. Results

After completing the preprocessing, feature extraction, and data splitting, we built three multi-classification models separately. These models were executed using TF-IDF feature extraction at multiple stages, incorporating N-gram analysis. This means we applied TF-IDF with unigrams across all models, saving the results of this feature. Additionally, we repeated the process with bigrams and trigrams, comparing the results achieved by various multi-classification machine learning algorithms (RF, DT, XGBoost) and two ensemble methods (voting and stacking). The RF model achieved the highest accuracy of 89.0% using TF-IDF with unigrams as a traditional ML classification. For the ensemble methods, specifically voting and stacking, the stacking classifier achieved an accuracy of 90.71%, while voting achieved 90% with unigram words. In addition, stacking outperformed, with an increase of 1% in the accuracy when using 2- and 3-gram features, surpassing the voting classifier. We used three metrics to evaluate the techniques: accuracy, F1-Score, and area under the curve (AUC), as illustrated in Table 3.

Table 3. Summary of comparative analysis of all classifiers based on performance evaluation.

Classifiers	**Random Forests**	**Decision Trees**	**XGBoost**	**Voting**	**Stacking**	**TF-IDF and N-Gram**
Accuracy	0.88 0.88 0.88	0.88 0.87 0.87	0.85 0.86 0.86	0.90 0.90 0.89	0.90 0.91 0.90	Unigram 2-Gram 3-Gram
F1-Score	0.89 0.89 0.89	0.89 0.89 0.89	0.86 0.86 0.86	0.90 0.90 0.90	0.90 0.90 0.90	Unigram 2-Gram 3-Gram
AUC	0.89 0.89 0.89	0.89 0.88 0.89	0.86 0.86 0.86	0.90 0.90 0.89	0.90 0.90 0.90	Unigram 2-Gram 3-Gram

Table 3 summarizes the results obtained from five classifiers: three multi-classification classifiers and two ensemble classifiers. Upon comparing the performance of multi-classification classifiers, Random Forest (RF) achieved the highest accuracy across all evaluation metrics, consistently maintaining the results while employing feature extraction with three different N-gram words. The Decision Tree (DT) classifier achieved similar performance to RF for most N-gram levels. XGBoost had lower accuracy initially; however, its performance slightly improved with the use of 2- and 3-grams. In terms of ensemble comparison, both classifiers demonstrated similar performance when using unigram words. However, stacking outperformed voting when using 2- and 3-grams. Additionally, employing bigrams with TF-IDF exhibited favorable performance in our experiment, as the results remained consistent across most N-gram tests and other evaluation metrics, ranging from 89% to 91% on the F1-Score and AUC when using bigrams. Stacking outperformed by 1% on the F1-Score, accuracy, and AUC when using unigrams. The voting classifier utilizes a hard voting technique, where the label agreed upon by the majority of algorithms in the classifier is used. This technique proved to be most efficient in our experiment. Considering all factors, our experiment showed that the ensemble models were outperformed, with multi-classification classifiers also surpassing traditional ML classifiers.

4.1. Evaluation Metrics

It is possible to assess the potential of each model using evaluation metrics to determine the classifiers' ability to classify five classes of cyberbullying in the dataset or the class of not cyberbullying. The efficacy of the models was evaluated by examining their respective evaluation metrics, including the accuracy, precision, recall, AUC, and the F-score from the confusion matrix, which were used to rank the models.

The accuracy of a model represents the fraction of its predictions that are correct out of the total predictions made by the model. The formula below provides an estimate using the accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

where:

TP = True positive.
FP = False positive.
FN = False negative.
TN = True negative.

The precision of a model enables the determination of the proportion of useful information among true positive (TP) and false positive (FP) data to identify a specific class.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

One definition of recall is the rate at which true positives are correctly predicted.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

F1-score is the harmonic mean of the precision and recall. Additionally, the F1-Score serves as a single measure that combines both the precision and recall.

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{7}$$

Another evaluation metric employed in our assessment is the area under the curve (AUC), which is scale-invariant and measures how well predictions are ranked compared to their correct values. The authors in [36] utilized this metric to evaluate the performance outcomes based on their experiment.

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

4.2. Comparison of Ensemble Models with Simple Traditional Models

Our aim in this paper was to develop a framework for detecting multiple classes of cyberbullying. This section discusses all the evaluation metrics for our approaches for both the ensemble classifiers that combined three multi-classifications models compared with two simple classifiers. The effectiveness of ensemble learning stems from its capacity to harness complementary learning processes with diverse strengths. In our experiment, we developed an automated system to detect cyberbullying tweets, surpassing the capabilities of traditional systems. Table 4 illustrates a comparison of the evaluations between our proposed ensemble models and simple traditional models.

Table 4. Summarizes the comparative analysis of all classifiers' performance.

Models	Accuracy	Precision	Recall	F1-Score	AUC
Multinomial NB	0.8055	0.8006	0.8058	0.8000	-
Logistic regression	0.8541	0.8604	0.8544	0.8566	-
Voting	0.9041	0.9069	0.9036	0.9045	0.9030
Stacking	0.9071	0.9085	0.9060	0.9063	0.9008

5. Discussion

The dataset utilized in our experiment is both recent and widely employed in various studies focused on detecting cyberbullying. Its recentness introduces new idioms associated with cyberbullying. Moreover, certain tweets present challenges in identifying the type of cyberbullying due to the presence of multiple words. Additionally, the dataset poses certain complexities, characterized by six target labels termed multiclass, further complicating the detection process. This section delves into the comparison of our framework with others that have explored the same dataset, demonstrating the enhancement in our performance. As depicted in Table 1, numerous experiments were conducted on the same dataset, each employing diverse methods for detection and prediction. This experiment [10] aimed to make a contribution by developing a system using an ensemble of transformer models, proposing a new framework in this field. The dataset comprises six labeled classes, prompting them to train their model in two scenarios: first, within five classes by excluding one labeled class and, second, within all six classes of the dataset. According to the experimental results, the model achieved higher accuracy when trained on five classes compared to six classes, attributed to the reduction in the number of classes and rows in the dataset. They utilized three deep learning models, namely RoBERTa, XLNet, and GPT2, combined into one Max-voting ensemble method. Their approach achieved an evaluation model with 85.25% accuracy, an 85.10% F1-Score, 85.02% precision, 85.25% recall, and an overall performance falling within the range of 85% for the classification of all classes in the dataset. We selected this study for comparison due to its similarity in approaches, involving ensemble techniques and encompassing all labels of the dataset in the second scenario. Consequently, our approach outperformed it, achieving a 90.71% accuracy, 90.63% F1-Score, 90.85% precision, and 90.60% recall.

Additionally, the authors in [11] introduced a sophisticated machine learning approach for deep analysis to detect cyberbullying in the dataset that we employed. In this study, they assessed the performance of five diverse machine learning models—LightGBM, XGBoost, Logistic Regression, Random Forest, and AdaBoost—using a textual feature extraction. They included all six classes of dataset, which consisted of over 47,000 tweets. Their analysis revealed that LightGBM outperformed the other models, achieving notable accuracy rates of 85.5%, precision rates of 84%, recall rates of 85%, and an F1-Score of 84.49%. We opted to contrast our study with this particular research due to its resemblance in its approaches, which involved utilizing machine learning techniques and incorporating all labels of the dataset. As a consequence, our methodology demonstrated superior performance, attaining 90.71% accuracy, a 90.63% F1-Score, 90.85% precision, and 90.60% recall. In our experiment, we implemented an ensemble technique that combined three commonly used classifiers, proving to be efficient for multiclass classification. In contrast, they employed traditional machine learning classifiers. Furthermore, we utilized TF-IDF as the word representation method, while they opted for textual features

The goal of this study was to develop a model that enhances system performance. We aimed to compare our model's approach with recent experiments that utilized the same dataset. Consequently, we selected all classes in the dataset for training the model, covering all types of cyberbullying present in these classes. Our experiment outperformed recent studies, achieving the highest results across all evaluation metrics and classifying entire datasets. We utilized ensemble learning to combine three classifiers previously employed for the individual classification and detection of cyberbullying tweets. These classifiers were integrated into a unified model using two ensemble learning techniques: voting and stacking. The stacking technique resulted in our model achieving 90.71% accuracy, while voting achieved 90% accuracy. For feature extraction, TF-IDF was employed with various N-grams, including unigrams, bigrams, and trigrams, among which unigrams demonstrated the best performance. TF-IDF offers several advantages, such as the ease of implementation, the simplicity in calculating document similarity, and robustness against common words. However, a limitation arose in its ability to capture the nuanced meanings between words. The authors in [15] achieved the highest accuracy using unigrams, a result consistent with our experiment. However, in [12], three-grams improved accuracy when employing the NN and SVM models. Therefore, our experiment suggests that considering the content of the text is crucial for accuracy improvement when using N-gram features. Combining multiple words as one token may impact the context of tweets, potentially confusing the model.

6. Conclusions

Cyberbullying is a complex issue, magnified by the widespread use of social media. In light of this, the automation of cyberbullying detection methods on social media platforms has become increasingly crucial. Our experiment contributes to this area by introducing an automated system designed to identify various types of cyberbullying on the Twitter platform. It is worth noting that the dataset was collected before Twitter's name changed to X. The study focused on exploring the effectiveness of ensemble multi-classification models for detecting diverse cyberbullying tweets. Our findings suggest that both voting and stacking ensemble classifier techniques outperformed recent experiments that aimed to detect cyberbullying using the same dataset. Moreover, our multi-classification classifiers demonstrated superior performance compared to traditional machine learning classifiers, achieving an increase of approximately 5% in accuracy across most models. We introduced three multi-classification models, namely Decision Trees (DTs), Random Forest (RF), and XGBoost, for detecting various classes of cyberbullying. Among these models, Decision Trees exhibited the highest accuracy at 89%, followed by Random Forest at 88%, and XGBoost at 86%. After testing these models, we applied ensemble techniques, which were voting and stacking classifiers, to combine these three models and enhance the performance. The results from both ensemble techniques revealed nearly identical outputs, with stacking exhibiting a marginal 1% improvement, signifying enhanced accuracy. Additionally, we conducted various N-gram experiments to optimize the performance. Among these, bigrams, particularly with TF-IDF feature extraction, demonstrated the most-promising results. We performed a detailed comparison between the two ensemble classifiers implemented in our study and juxtaposed our experimental results with recent studies in the field. This analysis employed well-established evaluation metrics, as detailed in this paper. In summary, our experiment showcased the superiority of ensemble models, with multi-classification classifiers outperforming traditional ML classifiers. Looking ahead, we intend to extend the application of our approach to different languages, such as Arabic, using multiclass datasets. This expansion will ensure the robust performance of our approach across diverse content.

Author Contributions: Conceptualization, A.F.A. and M.I.; methodology, A.F.A.; software, A.F.A.; validation, A.F.A. and M.I.; formal analysis, A.F.A. and M.I.; investigation, A.F.A. and M.I.; writing—original draft preparation, A.F.A.; writing—review and editing, A.F.A. and M.I.; visualization, A.F.A. and M.I; supervision, M.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset that used in this paper is available at [17].

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Boyd, D.; Golder, S.; Lotan, G. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In Proceedings of the 2010 43rd Hawaii International Conference on System Sciences, Honolulu, HI, USA, 5–8 January 2010; pp. 1–10.
2. Chapin, J. Adolescents and cyber bullying: The precaution adoption process model. *Educ. Inf. Technol.* **2016**, *21*, 719–728. [CrossRef]
3. Nobata, C.; Tetreault, J.; Thomas, A.; Mehdad, Y.; Chang, Y. Abusive language detection in online user content. In Proceedings of the 25th International Conference on World Wide Web, Montreal, QC, Canada, 11–15 April 2016; pp. 145–153.
4. Qureshi, K.A.; Sabih, M. Un-compromised credibility: Social media based multi-class hate speech classification for text. *IEEE Access* **2021**, *9*, 109465–109477. [CrossRef]
5. Qiu, S.; Xu, B.; Zhang, J.; Wang, Y.; Shen, X.; De Melo, G.; Long, C.; Li, X. Easyaug: An automatic textual data augmentation platform for classification tasks. In Proceedings of the Companion Proceedings of the Web Conference, Taipei, Taiwan, 20–24 April 2020; pp. 249–252.
6. Agrawal, S.; Awekar, A. Deep learning for detecting cyberbullying across multiple social media platforms. In *Advances in Information Retrieval, Proceedings of the 40th European Conference on IR Research, ECIR 2018, Grenoble, France, 26–29 March 2018*; Springer: Cham, Switzerland, 2018; pp. 141–153.
7. Ali, W.N.H.W.; Mohd, M.; Fauzi, F. Cyberbullying detection: An overview. In Proceedings of the 2018 Cyber Resilience Conference (CRC), Putrajaya, Malaysia, 13–15 November 2018; pp. 1–3.
8. Alam, K.S.; Bhowmik, S.; Prosun, P.R.K. Cyberbullying detection: An ensemble based machine learning approach. In Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 4–6 February 2021; pp. 710–715.
9. Muneer, A.; Fati, S.M. A comparative analysis of machine learning techniques for cyberbullying detection on twitter. *Future Internet* **2020**, *12*, 187. [CrossRef]
10. Ahmed, T.; Ivan, S.; Kabir, M.; Mahmud, H.; Hasan, K. Performance analysis of transformer-based architectures and their ensembles to detect trait-based cyberbullying. *Soc. Netw. Anal. Min.* **2022**, *12*, 99. [CrossRef]
11. Mahmud, M.I.; Mamun, M.; Abdelgawad, A. A deep analysis of textual features based cyberbullying detection using machine learning. In Proceedings of the 2022 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT), Alamein New City, Egypt, 18–21 December 2022; pp. 166–170.
12. Hani, J.; Mohamed, N.; Ahmed, M.; Emad, Z.; Amer, E.; Ammar, M. Social media cyberbullying detection using machine learning. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 703–707. [CrossRef]
13. Hasan, M.T.; Hossain, M.A.E.; Mukta, M.S.H.; Akter, A.; Ahmed, M.; Islam, S. A Review on Deep-Learning-Based Cyberbullying Detection. *Future Internet* **2023**, *15*, 179. [CrossRef]
14. Bhatt, C.M.; Patel, P.; Ghetia, T.; Mazzeo, P.L. Effective heart disease prediction using machine learning techniques. *Algorithms* **2023**, *16*, 88. [CrossRef]
15. Kadamgode Puthenveedu, S. Cyberbullying Detection Using Ensemble Method. Ph.D. Thesis, Carleton University, Ottawa, ON, Canada, 2022.
16. Ulus, C.; Wang, Z.; Iqbal, S.M.; Khan, K.M.S.; Zhu, X. Transfer Naïve Bayes Learning using Augmentation and Stacking for SMS Spam Detection. In Proceedings of the 2022 IEEE International Conference on Knowledge Graph (ICKG), Orlando, FL, USA, 30 November–1 December 2022; pp. 275–282.
17. Wang, J.; Fu, K.; Lu, C.T. Sosnet: A graph convolutional network approach to fine-grained cyberbullying detection. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 1699–1708.
18. Lee, C.S.; Cheang, P.Y.S.; Moslehpour, M. Predictive analytics in business analytics: Decision Tree. *Adv. Decis. Sci.* **2022**, *26*, 1–29.
19. Naeem, M.Z.; Rustam, F.; Mehmood, A.; Ashraf, I.; Choi, G.S. Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms. *PeerJ Comput. Sci.* **2022**, *8*, e914. [CrossRef]
20. Sarker, I.H. Machine learning: Algorithms, real-world applications and research directions. *SN Comput. Sci.* **2021**, *2*, 160. [CrossRef]
21. Qi, Z. The text classification of theft crime based on TF-IDF and XGBoost model. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 27–29 June 2020; pp. 1241–1246.
22. Ruta, D.; Gabrys, B. Classifier selection for majority voting. *Inf. Fusion* **2005**, *6*, 63–81. [CrossRef]

23. Rahman, M.M.; Islam, M.N. Exploring the performance of ensemble machine learning classifiers for sentiment analysis of COVID-19 tweets. In *Sentimental Analysis and Deep Learning: Proceedings of ICSADL 2021*; Springer: Singapore, 2021; pp. 383–396.
24. Alotaibi, Y.; Ilyas, M. Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things' Devices Security. *Sensors* **2023**, *23*, 5568. [CrossRef]
25. Wu, H.; Yuan, N. An Improved TF-IDF algorithm based on word frequency distribution information and category distribution information. In Proceedings of the 3rd International Conference on Intelligent Information Processing, Guilin, China, 19–20 May 2018; pp. 211–215.
26. Aizawa, A. An information-theoretic perspective of tf–idf measures. *Inf. Process. Manag.* **2003**, *39*, 45–65. [CrossRef]
27. Stamatatos, P.D. On the robustness of authorship attribution based on character n-gram features. *J. Law Policy* **2013**, *21*, 7.
28. Cheng, L.; Guo, R.; Silva, Y.; Hall, D.; Liu, H. Hierarchical attention networks for cyberbullying detection on the instagram social network. In Proceedings of the 2019 SIAM International Conference on Data Mining, Calgary, AB, Canada, 2–4 May 2019; pp. 235–243.
29. Zhou, H. Research of text classification based on TF-IDF and CNN-LSTM. *J. Phys. Conf. Ser.* **2022**, *2171*, 012021. [CrossRef]
30. Al-Hashedi, M.; Soon, L.K.; Goh, H.N. Cyberbullying detection using deep learning and word embeddings: An empirical study. In Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems, Bangkok, Thailand, 23–25 November 2019; pp. 17–21.
31. Younas, M.; Wakil, K.; Jawawi, D.N.; Shah, M.A.; Ahmad, M. An automated approach for identification of non-functional requirements using Word2Vec model. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10*, 539–547. [CrossRef]
32. Kowsari, K.; Jafari Meimandi, K.; Heidarysafa, M.; Mendu, S.; Barnes, L.; Brown, D. Text classification algorithms: A survey. *Information* **2019**, *10*, 150. [CrossRef]
33. Sham, N.M.; Mohamed, A. Climate change sentiment analysis using lexicon, machine learning and hybrid approaches. *Sustainability* **2022**, *14*, 4723. [CrossRef]
34. Alduailaj, A.M.; Belghith, A. Detecting arabic cyberbullying tweets using machine learning. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 29–42. [CrossRef]
35. Rustam, F.; Khalid, M.; Aslam, W.; Rupapara, V.; Mehmood, A.; Choi, G.S. A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis. *PLoS ONE* **2021**, *16*, e0245909. [CrossRef]
36. Alalwany, E.; Mahgoub, I. Classification of Normal and Malicious Traffic Based on an Ensemble of Machine Learning for a Vehicle CAN-Network. *Sensors* **2022**, *22*, 9195. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.