DOI: 10.24996/ijs.2023.64.9.40

Iraqi Journal of Science

ISSN: 0067-2904

Comparing the Random Forest vs. Extreme Gradient Boosting using Cuckoo Search Optimizer for Detecting Arabic Cyberbullying

Marwa Q. Saadi*[*] , Ban N. Dhannoon

Department of Computer Science, College of Science, Al-Nahrain University, Baghdad, Iraq

Received: 5/6/2022 Accepted: 2/11/2022 Published: 30/9/2023

Abstract

Cyberbullying is one of the major electronic problems, and it is not a new phenomenon. It was present in the traditional form before the emergence of social networks, and cyberbullying has many consequences, including emotional and physiological states such as depression and anxiety. Given the prevalence of this phenomenon and the importance of the topic in society and its negative impact on all age groups, especially adolescents, this work aims to build a model that detects cyberbullying in the comments on social media (Twitter) written in the Arabic language using Extreme Gradient Boosting (XGBoost) and Random Forest methods in building the models. After a series of pre-processing, we found that the accuracy of classification of these comments was 0.861 in XGBoost, and 0.849 in Random Forest. Then the results of this model were improved by using one of the optimization algorithms called cuckoo search to adjust the parameters in two methods. The results are improved clearly in the random forest method, which obtained results similar to the extreme gradient boosting method, with a value of 0.867.

Keywords: Cyberbullying**,** XGBoost**,** Random Forest**,** Machine Learning**,** Cuckoo Search**.**

مقارنة الغابة العشوائية مقابل تعزيز التدرج الشديد باستخدام محسِّن البحث الوقواق لاكتشاف التنمر عبر الإنترنت باللغة العربية

مروة قاسم سعدي*، بان نديم ذنون

قسم الحاسبات, كلية العلوم , جامعة النهرين، بغداد، العراق

الخلاصة

يعتبر التنمر الإلكتروني من المشاكل الإلكترونية الكبرى وليست ظاهرة جديدة، لكنها كانت موجودة بالشكل التقليدي قبل ظهور الشبكات الاجتماعية، وللتنمر الإلكتروني عواقب عديدة، منها الحالات العاطفية والفسيولوجية، مثل الاكتئاب والقلق. ونظرا لانتشار هذه الظاهرة ولأهمية الموضوع في المجتمع وتأثيره السلبي على جميع الفئات العمرية وخاصة المراهقين، فإننا نهدف من خلال هذا العمل إلى بناء نموذج يكتشف التنمر الإلكتروني في تعليقات أحد مواقع التواصل الاجتماعي (تويتر) مكتوبة باللغة العربية باستعمال أسلوب تعزيز التدرج الشديد (XGBoost) والغابة العشوائية في بناء النماذج. بعد سلسلة من المعالجة المسبقة، وجدنا أن دقة تصنيف هذه التعليقات كانت 0.861 في تعزيز التدرج الشديد و 0.849 في الغابة العشوائية. تم تحسين النتائج بشكل واضح

[*]Email: marwasaadi059@gmail.com

باستعمال احدى خوارزميات التحسين هي خوارزمية بحث الوقواق وطريقة الغابة العشوائية، والتي حصلت على نتائج مماثلة لطريقة التعزيز القصوى للتدرج بقيمة 0.867.

1. Introduction

The internet is considered one of the daily necessities in the lives of all people. Moreover, access to the Internet is easy without any limitations on distance and time. Therefore, getting information is very easy for anyone [1]. The means of access to the Internet provide an opportunity for all social media to expand the extent of their use among all Internet users, especially teenagers, who consider social media a recent trend to occupy their spare time with activities and events in the electronic space. Although the Internet is considered a harmless thing for users, the flexibility that exists on the Internet may be a major factor slowly The emergence of problems such as cyberbullying [2], which has recently been considered a health and national issue [3], [4]. Cyberbullying cannot be considered a new phenomenon in the world that has emerged recently. Rather, it existed, but it existed in the traditional form before the emergence of social networks, for example, face-to-face between the bully and the victim [5] and [6]. This is the opposite of cyberbullying because the incident is in a broader field, which is cyberspace, so there is difficulty in detecting the bullies because there is ambiguity in identifying the parties involved in the incident [7] and [8]. Among the most important consequences of electronic bullying are emotional and physiological, for example, depression, anxiety, panic attacks, lack of self-esteem, and low self-confidence [9] and [10]. However, the harshest and most serious consequence of cyberbullying is suicide [11] and [12]. Given the spread of this phenomenon and the importance of the subject to society, we aim through this work to build a model to detect cyberbullying in the Arabic comments on social media by a machine learning method and improve the results of this model through one of the optimization algorithms, cuckoo search (CS).

2. Related Work

In [13], the goal of this research study is to collect a dataset of tweets and evaluate and categorize them using various machine learning methods. The performance of different classifiers is shown to vary depending on the size of the data collection. The findings were that Naive Byes and ID3 performed better with balanced data sets. With imbalanced data sets, different classifiers (K-NN, Decision Tree, RF) performed better. DT and K-NN show superior results with imbalanced datasets. With balanced datasets, the highest accuracy is 39.4 %, but with unbalanced datasets, it is 82.7 %.

In [14], rhetorical strategies in Arabic are defined as forms of linguistic expression that communicate thoughts and sentiments through written or spoken texts. They developed an XGBoost classifier for classifying multi-layer Arabic pictorial texts. This specialized research point is in the Arabic language. This Arabic picture collection was used to design, train, and test the XGBoost Workbook (AFC). The XGBoost rated as F1 has an 88 % performance rating.

In [15], they uses machine learning to detect fear in response to government attempts to combat the epidemic based on social media input. Sentiment analysis will be used in this idea to identify anxiety based on positive and negative input from Internet users. K-NN, Bernoulli, Decision Tree, SVM, RF, and XGBoost are some of the machine learning algorithms used. The sample data utilized was obtained via crawling YouTube comments. The highest accuracy was a random forest with vector number extraction advantage and TF-IDF, with 84.99 % and 82.%, respectively. K-NN is the most accurate test, whereas XG-Boost has the best recall.

[16] presented Arabic Cyberbullying Detection Using Arabic Sentiment Analysis. The researchers used tools such as AraBully Keywords to collect data from Twitter for their study. The total number of tweets was 17748, and the researcher used SVM, WEKA, and Python compilation tools to work in the compilation phase after pre-processing. They found that when using Light Stemmer, WEKA accurately ranked 15,252.6312 tweets (85.49 %), and when using ArabicStemmerKhoja, WEKA ranked 15,154 tweets (85.3843%), but Python correctly ranked 14,908.32 tweets (84.03 %).

In [17], the researchers present a method for categorizing textual tweets in the Arabic language into five separate groups based on their linguistic traits and content. The Support Vector Machine (SVM), Gaussian Nave Bayes (GNB), and Random Forest (RF) were tested. When employed with stemming and term frequency-inverse document frequency (TF-IDF), the RF and the SVM with radial basis function (RBF) kernel fared similarly well statistically, with macro-F1 scores ranging from 98.09% to 98.14%.

In [18], the purpose of this research is to label a news item automatically based on its lexical properties. Two huge datasets were compiled from several Arab news sources. They utilized a collection of 10 shallow learning classifiers to look at the single-label data set. In addition, all of the classifiers tested, incorporated an aggregate model that used the majority voting approach. The classifiers' performance in the first data set varied from 87.7% (AdaBoost) to 97.9% in both shallow learning and multi-tag deep learning algorithms. The accuracy of XGBoost was 84.7 %, while the accuracy of logistic regression was 81.3 %. The accuracy of the second dataset was 94.85 % after it was examined by various deep learning neural networks. The Convolutional Gated Recurrent Unit (CGRU) was shown to be the top multi-label classifier**.**

3. Data Collection

The raw data set in this work was obtained from [16] and stored in a "comma-separated value" (CSV). It includes comments on Twitter written in Arabic, some of which use words that refer to cyberbullying, such as ugly descriptions and offensive words in expressing an opinion or describing people with negative qualities, and others using words that refer to cyberbullying. Positive in praising a specific content or nice words in the grace of religions or praising the performance of well-known personalities. The total number of tweets is 17,748, the number of cyber-bullying tweets is 14,178, and the non-cyber-bullying tweets are (3,570). So, the dataset is considered an unbalanced dataset.

4. Imbalanced Class Distribution

Imbalanced class distribution means the distribution of the classes that are not equal. The machine learning algorithm works well when the classes have almost the same number of instances [19]. There are various techniques for solving imbalanced classes, one of which is downsampling, done by randomly picking the number of instances in the highest class and making it semi-equal to the lowest class. The second technique is to resample only the training set and make it over-sampling [19] and [20]. Upsampling is done using the Synthetic Minority Oversampling Technique (SMOTE) [21]. We combine up-sampling and down-sampling together as another technique.

5. Methodology and Experiments

Detecting cyberbullying involves a binary classification task related to comments and distinguishing whether it is cyberbullying (denoted by C) or non-cyberbullying (denoted by N).

The general steps of the proposed model start by collecting the dataset and rebalancing it through three techniques (downsampling, upsampling, and an up-down sampling technique by combining the first two techniques). Data is preprocessed using seven steps:

1- Remove duplicated: This process deletes any duplicated tweets in the dataset.
2- Tokenization: It is the process of splitting up tweets into words called tokens, separated by commas [22].
3- Normalization: This is the process of transforming a text into a canonical form by removing noises, such as dates, whitespaces, abbreviations, acronyms, and diacritics [23], [24].
4- Stop word removal: This process is used to filter out unnecessary data by converting it into an accepted form. The removal of this unnecessary data does not affect the general meaning of the text [25]. This work obtained a list of Arabic stop words from [16] and store it in an excel sheet (.xlsx).
5- Stemming: It is a normalization technique in which a list of distinct words is converted into shortened root words to eliminate redundancy. This is done by removing their affixes and suffixes. Root stemming and light stemming are two types of stemming in the Arabic language [26].
6- Lemmatization: It is used to reduce words to their base representation by returning them to their meaning in proper form considering their morphological analysis.
7- Padding: Since some sentences are long and others are short, post-zeros padding with a maximum length equal to the longest sentence in the dataset is used to equalize the length of sentences [27]. Table (1) shows examples for each step in pre-processing.

Table 1: Example of Pre-processing

Pre-processing	Before Pre-processing	After Pre-processing
Remove duplicated	The number of tweets is: 17748	after removing duplicated: 17726
Tokenization	ااانت غبي بمعنى غبي	ااانت, غبي , بمعنى, غبي
Normalization	ااانت , غبي , بمعنى, غبي	انت , غبي , بمعنى , غبي
Remove StopWord	انت , غبي , بمعنى , غبي	غبي بمعنى غبي
Stemming and Lemmatization	فاهمون	Light : فاهم , Root : فهم, Lemma : فَهْم

The pre-processed data is split into training and testing. Then the feature extractor method is applied. The resulting features are classified and evaluated using the random forest and XGBoost methods before and after the Cuckoo search optimizer method. Figure (1) shows the proposed model used in this work.

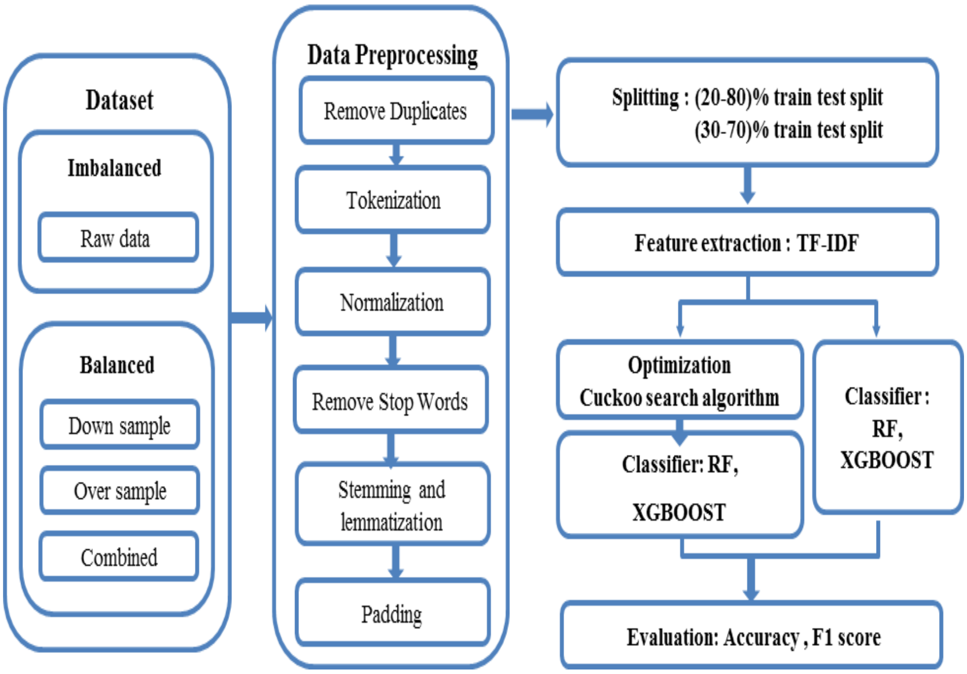


Figure 1: The proposed model

5.1. Feature extraction

Feature extraction is considered an important step in data mining and information retrieval. It transforms unstructured main text into structured information that the computer can distinguish and process [28]. This work applied the time-frequency–inverse document frequency (TF_IDF) for the feature extraction process as shown in eq. (1), which utilizes the TF and IDF of each word in the document after getting vectors for all words. In eq. (2), TF (w) signifies the frequency of the word w in the document, count (w) and count (w_n) denote the number of samples including the word w in the dataset, and n denotes the number of samples containing the word w in a corpus, respectively [29]. IDF (w) denotes the inverse file frequency of the word w in equation (3).

$$TF - IDF(w) = \mathrm{TF} * \mathrm{IDF} \tag{1}$$

$$TF(w) = \frac{count(w)}{\sum count(w_n)} \tag{2}$$

$$IDF(w) = log\left(\frac{N}{df}\right) \tag{3}$$

5.2. Machine Learning Algorithm

Machine learning (ML) refers to a computer's ability to teach itself how to make decisions using both available data and experience [30]. The available data is called “training data”. Decisions to be made in ML are either classification or prediction of new things or data. A computer classifies new data or objects based on learning algorithms. If the training data is classified by human experts, then all algorithms that depend on this type of data are called supervised learning algorithms [31]. There are many machine learning algorithms available, and the algorithms used in this work will be mentioned as follows:

1- **Decision Tree (DT):** It is one of the most well-known classification algorithms in supervised learning. Its work is to build a prediction tree structure using information-entropy concepts using annotated training datasets [32]. DT: Learners use a set of labelled data and classify it using a command-and-conquer approach. Each tree consists of leaves representing a classification class and arches representing a feature checked from the training data [33].

2- **Random Forest (RF):** is an ensemble learning algorithm [34], proposed by Breiman (2001) [35]. It is composed of multiple independent DTs that are trained independently on a random subset of data. RF builds many tree models by using the training data. These trees can be used for prediction in later phases. This type of ML algorithm is different from DT. It does not suffer from the overfitting problem that is found in DT [36]. Various training set samples are randomly created from the main training set using a bootstrap sampling method in its process. DT operations are then conducted on the new training sets while constructing the DT. The DT is created by choosing a random column variable and merging it with a random row observation. All these sets of DT together make (RF) [35]. RF can be used in classification and regression problems [37].

3- **XGBoost (Extreme Gradient Boosting):** it is an ensemble learning algorithm [38] proposed by Dr. Chen in 2016 [39] and considered one of the effective ML algorithms based on DT algorithms as its central unit. It is considered more accurate than a single DT algorithm. The XGBoost algorithm uses successive dataset training processes to merge weak predictors (DT) and achieve more strong predictors [35]. XGBoost is a robust ML algorithm for both classification [40] and regression [41].

5.3. Evaluation Metric

Minimizing the error rate of classification performance over the validation set of given training data [42] is the aim of the proposed algorithm. This performance was assessed by its accuracy and the f1-score [43]. Table (2) shows their formula.

Table 2 : Evaluation Metric

Metric Name	Calculation
Recall	$TP/(TP + FN)$
Precision	$TP/(TP + FP)$
F1 Score	$2 * (Precision * Recall)/(Precision + Recall)$
Accuracy	$(TP + TN)/(TP + FN + TN + FP)$

5.4. Hyperparameter Optimization

The handling of hyperparameters in machine learning allows for controlling the performance of the algorithm in the dataset. Hyperparameter optimization adjusts a set of hyperparameters of the learning algorithm to improve the performance of the ML model [44]. Because of the different types of hyperparameters, each ML algorithm has a different tuning process for its hyperparameters [45].

5.5. Cuckoo Search Algorithm

Cuckoo search is inspired by the bird cuckoo, which can be defined as a meta-heuristic algorithm. This kind of bird puts its eggs in the nest of another host bird because it never builds its own nest. The host bird recognizes the eggs that are not its own eggs and either disposes of the eggs from its nest or simply removes them from its nest and builds a new nest. Each egg in a nest represents a solution, and a new and good solution is represented by the cuckoo egg. The obtained solution is a new solution based on the existing one with some differences [46]. There are three main rules in CS [47].

First: each cuckoo selects a nest randomly to puts eggs in.
Second: the number of available host nests is constant, and nests with the top-quality eggs will carry over to the next generations.

Third: if the host bird distinguishes the cuckoo egg, it can dispose of the egg or leave the nest, and build a new one.

There is a constant number of host nests, and the probability that an egg laid by a cuckoo is distinguished by the host bird is pc ∊ [0, 1]. Figure (2) shows a change in the value of the parameters using a cuckoo search.

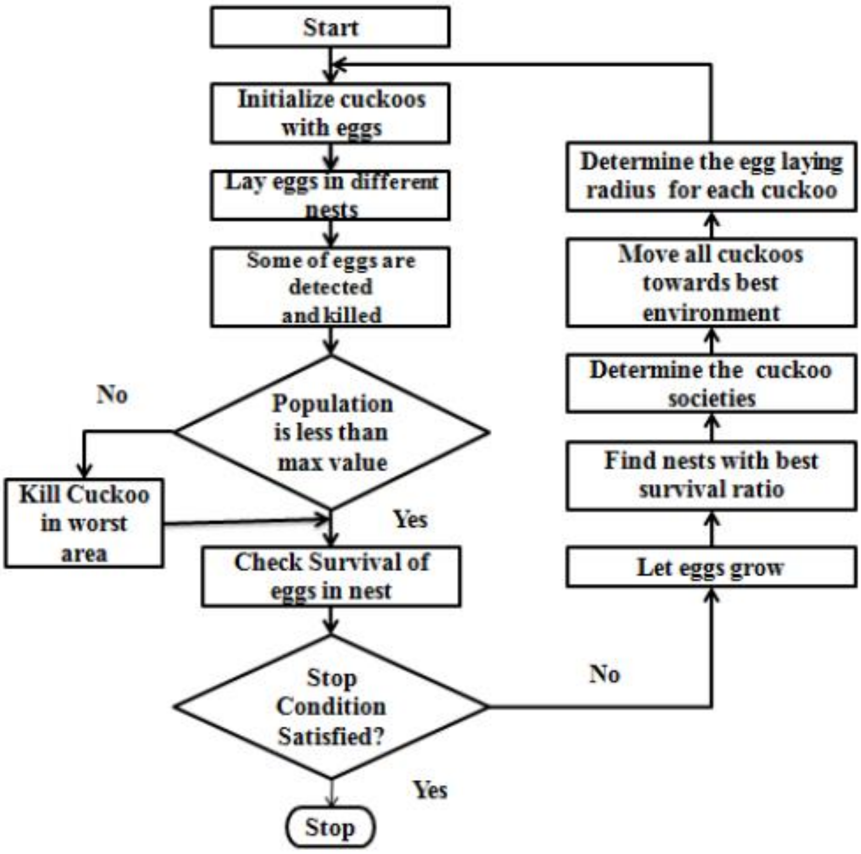

Start
Initialize cuckoos with eggs
Lay eggs in different nests
Some of eggs are detected and killed
Population is less than max value
No
Kill Cuckoo in worst area
Yes
Check Survival of eggs in nest
Stop Condition Satisfied?
No
Yes
Stop
Let eggs grow
Find nests with best survival ratio
Determine the cuckoo societies
Move all cuckoos towards best environment
Determine the egg laying radius for each cuckoo

Figure 2: Change Value of the Parameters Using a Cuckoo Search

.

6. Hyperparameters in XGBoost and RF

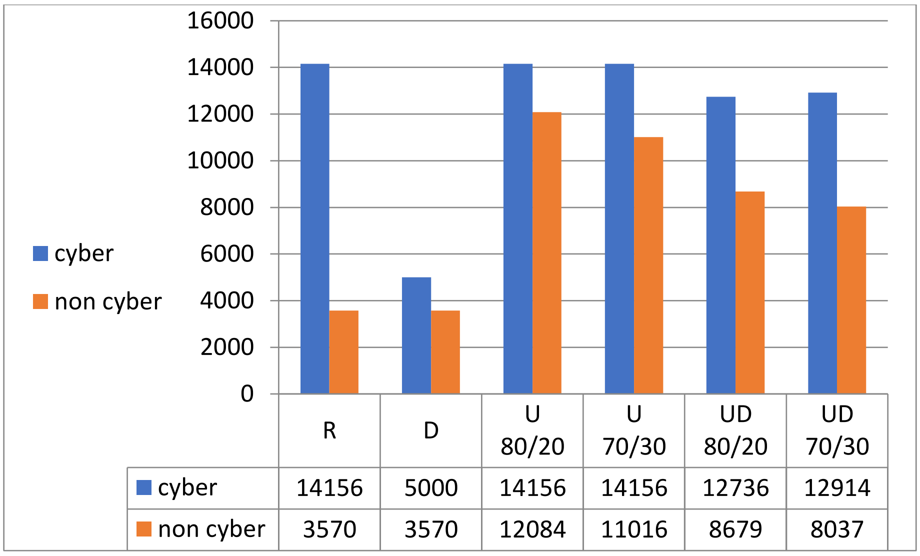
XGBoost and RF classifiers have many kinds of hyperparameters that need to be determined and applied to the optimization algorithm (CS). Table 3 shows the RF and XGBoost classifier's hyperparameters, range, and default values [48], [40].

Table 3: Hyperparameters in XGBoost and RF

Classifier	Parameter	Range	Default
XGBoost	Eta	[0,1]	0.3
	Gamma	$[0, \infty)$	0
	Min- child weight	$[0, \infty)$	1
	Scale -pos weight	Unlimited	1
RF	Max depth	Unlimited	none

7. Experimental Setup

Before feature extraction and classification, the dataset must be split into (80-20) or (70-30)% train-test datasets. This division gives the ability to model for training and testing on an unseen dataset. Each set of training and tests was applied separately to the feature extraction process and then to the classifier stage. Figure (3) shows the five corpora corpus used in experiments and the number of instances of the four corpora: Raw dataset (R), Downsampling dataset (D), Upsampling dataset (U), and Up+Down sampling Dataset (UD).



	R	D	U 80/20	U 70/30	UD 80/20	UD 70/30
■ cyber	14156	5000	14156	14156	12736	12914
■ non cyber	3570	3570	12084	11016	8679	8037

Figure 3: Main Corpora and Number of Instances

All experiments done in this work were dependent on stratified 4-fold cross-validation in the case of 70–30% train test split and on 5-fold cross-validation in the case of 80–20% train test split by using an inner loop (for the training set) and an outer loop (for the main dataset).

8. Results and Discussion

A total of 24 sets of experiments have been trained and tested on different corpora using two classifiers. Tables (4) and (5) show the performance of classification and metric evaluation (accuracy, F1-score) on different corpora for XGBoost and RF, respectively. The classification operation in Table (4) was done by using an XGBoost classifier with four corpora in light, root stemming, and lemmatization, so the highest accuracy was with the root stemming and R corpus. While almost the lowest accuracy with the D corpus, this means that downsampling does not work well with this type of dataset and classifier. While in the table (5), the classification is done by using an RF classifier with four corpora in light, root stemming, and lemmatization, so the highest accuracy is with the root stemming and UD corpora. While R corpus and D have nearly the lowest accuracy.

Table 4: XGBoost Classifier on Average of 5-Fold Cross-Validation

		Train _ Test split 80- 20 %				**Train _ Test split 70- 30 %**			
Stem	Corpus	Train	Val	Test	F1	Train	Val	Test	F1
Light	R	0.915	0.829	0.858	0.910	0.915	0.824	0.860	0.913
	U	0.930	0.825	0.851	0.909	0.932	0.829	0.849	0.899
	D	0.923	0.815	0.816	0.830	0.925	0.812	0.806	0.820
	UD	0.933	0.820	0.847	0.897	0.930	0.828	0.839	0.891
Root	R	0.962	0.862	0.861	0.914	0.964	0.860	0.858	0.910
	U	0.962	0.847	0.856	0.907	0.965	0.846	0.851	0.908
	D	0.962	0.814	0.798	0.826	0.964	0.810	0.796	0.823
	UD	0.960	0.861	0.854	0.908	0.963	0.848	0.854	0.906
Lemma	R	0.957	0.860	0.816	0.831	0.960	0.856	0.816	0.835
	U	0.904	0.800	0.803	0.816	0.968	0.826	0.802	0.816
	D	0.897	0.808	0.817	0.827	0.899	0.806	0.815	0.828
	UD	0.953	0.846	0.843	0.897	0.957	0.845	0.846	0.899

Table 5: RF classifier on average of 5-fold cross-validation

		Train _ Test split 80- 20 %				**Train _ Test split 70- 30 %**			
Stem	Corpus	Train	Val	Test	F1	Train	Val	test	F1
Light	R	0.875	0.812	0.815	0.822	0.878	0.811	0.813	0.820
	U	0.941	0.817	0.819	0.825	0.939	0.812	0.818	0.827
	D	0.932	0.822	0.813	0.820	0.948	0.821	0.823	0.836
	UD	0.938	0.820	0.832	0.847	0.937	0.815	0.831	0.842
Root	R	0.845	0.811	0.805	0.817	0.847	0.810	0.812	0.818
	U	0.940	0.813	0.822	0.833	0.948	0.820	0.823	0.837
	D	0.946	0.812	0.805	0.809	0.935	0.817	0.807	0.816
	UD	0.938	0.818	0.848	0.904	0.943	0.822	0.849	0.905
Lemma	R	0.907	0.817	0.810	0.819	0.876	0.814	0.813	0.819
	U	0.965	0.831	0.834	0.850	0.968	0.826	0.835	0.841
	D	0.968	0.822	0.822	0.836	0.974	0.814	0.813	0.820
	UD	0.959	0.826	0.839	0.859	0.947	0.820	0.841	0.890

After completing the classification process and before starting the optimization process, the range for each hyperparameter in both classifiers must be determined. Table (6) shows the range for each parameter in both XGBoost and RF.

Table 6: The Range of Hyperparameters

Classifier	**Parameter**	**Rang in our work**
XGBoost	Eta	[0,1]
	Gamma	[0,10]
	Min child wight	[0,10]
	Scale pos weight	[0.5, 1.6]
RF	Max depth	[100,1000]

After selecting a range of hyperparameters in both classifiers, the optimization process starts by applying the CS algorithm with R and UD corpora, since they show higher accuracy with a smaller number of iterations

Table (7) shows that the highest accuracy in the XGBoost classifier applied to 80% of the training data were achieved on the R corpus with lemmatization. It has a scale-pos-weight of (1.171), a gamma of (1.665), a min-child-weight of (0.268), and an Eta of (0.735), with the best accuracy on 70% of the training data achieved on the R corpus with a light stemmer. As shown in Table (8), while in the RF classifier, the highest accuracy was achieved in the UD corpus in both 80% and 70% of the training data with light stemming, max depth (1000), and (879), respectively.

Table 7: Optimization Stage with XGBoost Classifier

Corpus	Stem	iterations	Scale -post-weight	Gamma	Min-child-weight	Eta	F1 score	Accuracy
R	Light	18	0.500	0	0.288	0.841	0.912	0.863
	Root	5	1.082	0.034	0	0.439	0.916	0.864
	Lemma	10	1.171	1.665	0.268	0.735	0.917	0.867
UD	Light	17	0.500	2.072	0.611	0.366	0.907	0.855
	Root	13	0.500	1.160	0.755	0.328	0.912	0.860
	Lemma	15	0.536	2.253	0	0.286	0.916	0.866
R	Light	9	1.597	0.293	0	0.571	0.913	0.864
	Root	11	1.136	4.259	0.984	0.882	0.915	0.863
	Lemma	9	1.409	1.930	0	0.248	0.910	0.859
UD	Light	13	0.500	1.996	0.116	0.502	0.908	0.857
	Root	11	0.538	1.420	0	0.174	0.911	0.857
	Lemma	13	0.804	0.642	0.369	0.457	0.910	0.858

Following all of the previous experiments with various corpora and comparisons with previous works, it has been discovered that the best accuracy is achieved on XGBoost without using the cuckoo search optimizer, as shown in Table (9). Cuckoo search improves the accuracy of both classifiers, and they achieve the same accuracy. The RF responded greatly to the CS optimizer, raising its accuracy level to equal the XGBoost accuracy.

Table 8: Optimization Stage with RF Classifier

		Train-Test Split 80- 20 %				Train-Test Split 70- 30 %			
Corpus	Stem	Iterations	Max depth	F1 score	Accuracy	Iterations	Max depth	F1 score	Accuracy
R	Light	13	799	0.892	0.822	10	1000	0.894	0.821
	Root	10	000	0.913	0.858	4	768	0.911	0.856
	Lemma	5	941	0.899	0.828	8	940	0.894	0.820
UD	Light	19	000	0.914	0.867	9	879	0.910	0.860
	Root	18	968	0.912	0.858	13	752	0.910	0.856
	Lemma	19	848	0.911	0.861	14	1000	0.906	0.854

Table 9: Comparison with Previous Work

	Classifier	Corpus	Accuracy
Previous work **[11]**	SVM	R	84.04 %
This work	RF	UD	84.9 %
	RF+ CS	UD	86.7 %
	XGBoost	R	86.1 %
	XGBoost+ CS	R	86.7%

9. Conclusion

In this work, the unbalanced dataset problem was solved by producing three corpora, U, D, and UD, for upsampling, downsampling, and up-down sampling, respectively. The D corpus almost shows less accuracy with the XGBoost classifier since much information was missing during the rebalancing operation.

In the RF classifier, the R corpus shows the worst accuracy, while the UD corpus shows the best accuracy. This indicates that the random forest algorithm is inefficient with unbalanced datasets. In the XGBoost classifier, the R corpus shows the best accuracy, which indicates the resistance of the XGBoost algorithm to the unbalanced dataset. The Cuckoo search optimizer improves RF and XGBoost accuracy to 0.867 for both classifiers.

References

[1] 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 International Conference on Information and Communication Technology (ICICTM)*, pp. 246-250, May 2016.

[2] C. Van Hee *et al.*, "Automatic detection of cyberbullying in a social media text," *PloS one*, vol. 13, no. 10, pp.1-22, Oct 2018.

[3] M. A. Al-Garadi, K. D. Varathan, and S. D. Ravana, "Cybercrime detection in online communications: The experimental case of cyberbullying detection in the Twitter network," *Computers in Human Behavior*, vol. 63, pp. 433-443, May 2016.

[4] J.-M. Xu, K.-S. Jun, X. Zhu, and A. Bellmore, "Learning from bullying traces in social media," in *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pp. 656-666, Jun 2012.

[5] C. Salmivalli, "Bullying and the peer group: A review," *Aggression and violent behavior*, vol. 15, no. 2, pp. 112-120, Sep 2010.

[6] J. J. Dooley, J. Pyżalski, and D. Cross, "Cyberbullying versus face-to-face bullying: A theoretical and conceptual review," *Zeitschrift für Psychologie/Journal of Psychology*, vol. 217, no. 4, pp. 182-188, Jan 2009.

[7] D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, "Detection of harassment on web 2.0," *Proceedings of the Content Analysis in the WEB*, vol. 2, pp. 1-7, Apr 2009.

[8] T. K. Chan, C. M. Cheung, and Z. W. Lee, "Cyberbullying on social networking sites: A literature review and future research directions," *Information & Management*, vol. 58, no. 2, pp.1-65, Dec 2021.

[9] J. Hani, N. Mohamed, M. Ahmed, Z. Emad, E. Amer, and M. Ammar, "Social media cyberbullying detection using machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, Jan 2019.

[10] R. M. Kowalski and S. P. Limber, "Psychological, physical, and academic correlates of cyberbullying and traditional bullying," *Journal of adolescent health*, vol. 53, no. 1, pp. S13-S20, Jul 2013.

[11] A. Saravanaraj, J. Sheeba, and S. P. Devaneyan, "Automatic detection of cyberbullying from Twitter," *International Journal of Computer Science and Information Technology & Security (IJCSITS)*, vol.6, no.6, pp. 26-30, Nov-Dec 2016.

[12] D. Chatzakou, I. Leontiadis, J . Blackburn, ED. Cristofaro, G. Stringhini, A. Vakali, and N.Kourtellis, "Detecting cyberbullying and cyberaggression in social media," *ACM Transactions on the Web (TWEB),* vol. 13, no. 3, pp. 1-51, Oct 2019.

[13] A. Alshamsi, R. Bayari, and S. Salloum, "Sentiment analysis in English texts," *Advances in Science, Technology and Engineering Systems Journal,* vol. 5, no. 6, pp. 1683-1689, Dec 2020.

[14] N. S. Elmitwally, "Building A Multi-class XGBoost Model for Arabic Figurative Language," in *2020 2nd International Conference on Computer and Information Sciences (ICCIS),* pp. 1-4, Oct 2020.

[15] S. Saifullah, Y. Fauziah, and A. S. Aribowo, "Comparison of machine learning for sentiment analysis in detecting anxiety based on social media data," *arXiv preprint arXiv:2101.06353,* pp. 1-10, Jan 2021.

[16] S. Almutiry and M. Abdel Fattah, "Arabic CyberBullying Detection Using Arabic Sentiment Analysis," *The Egyptian Journal of Language Engineering,* vol. 8, no. 1, pp. 39-50, Apr 2021.

[17] S. M. Alzanin, A. M. Azmi, and H. A. Aboalsamh, "Short text classification for Arabic social media tweets," *Journal of King Saud University-Computer and Information Sciences,* vol.34, no.9, pp. 6595-6604, Oct 2022.

[18] H. El Rifai, L. Al-Qadi, and A. Elnagar, "Arabic text classification: the need for multi-labeling systems," *Neural Computing and Applications,* vol. 34, no. 2, pp. 1135-1159, Jun 2022.

[19] B. A. Talpur and D. O'Sullivan, "Cyberbullying severity detection: A machine learning approach," *PloS one,* vol. 15, no. 10, pp. 1-19, Oct 2020.

[20] N. Qazi, "Effect of Feature Selection, Synthetic Minority Over-sampling (SMOTE) And Under-sampling on Class imbalance Classification (2012)," ed.

[21] R. E. Al Mamlook, T. Z. Abdulhameed, R. Hasan, H. I. Al-Shaikhli, I. Mohammed, and S. Tabatabai, "Utilizing machine learning models to predict the car crash injury severity among elderly drivers," in *2020 IEEE International Conference on Electro Information Technology (EIT),* pp. 105-111, Jul 2020.

[22] Y. A. Alhaj, M. A. Al-qaness, A. Dahou, M. Abd Elaziz, D. Zhao, and J. Xiang, "Effects of light stemming on feature extraction and selection for Arabic documents classification," in *Recent Advances in NLP: The Case of Arabic Language*, Springer, vol. 874, pp. 59-79, 2020

[23] L. S. Larkey, L. Ballesteros, and M. E. Connell, "Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis," in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 275-282, Aug 2002.

[24] D. H. Abd, A. T. Sadiq, and A. R. Abbas, "Political articles categorization based on different naïve Bayes models," in *International Conference on Applied Computing to Support Industry: Innovation and Technology,* Springer, vol. 1174, pp. 286-301, Sep 2019.

[25] N. Shah and S. Mahajan, "Document clustering: a detailed review," *International Journal of Applied Information Systems,* vol. 4, no. 5, pp. 30-38, Oct 2012.

[26] M. Aljlayl and O. Frieder, "On Arabic search: improving the retrieval effectiveness via a light stemming approach," in *Proceedings of the eleventh international conference on Information and knowledge management,* pp. 340-347, Nov 2002.

[27] N. A. Hamzah and B. N. Dhannoon, "The Detection of Sexual Harassment and Chat Predators Using Artificial Neural Network," *Karbala International Journal of Modern Science,* vol. 7, no. 4, pp. 301-312, Dec 2021.

[28] V. Singh, B. Kumar, and T. Patnaik, "Feature extraction techniques for handwritten text in various scripts: a survey," International Journal of Soft Computing and Engineering (IJSCE), vol. 3, no. 1, pp. 238-241, 2013.

[29] T. A. Wotaifi and B. N. Dhannoon, "Improving Prediction of Arabic Fake News Using Fuzzy Logic and Modified Random Forest Model," Karbala International Journal of Modern Science, vol. 8, no. 3, pp. 477-485, 2022.

[30] Q. Liu, J. Wang, D. Zhang, Y. Yang, and N. Wang, "Text features extraction based on TF-IDF associating semantic," in 2018 IEEE 4th International Conference on Computer and Communicat*ions (ICCC)*, pp. 2338-2343, Dec 2018.

[31] J. Alzubi, A. Nayyar, and A. Kumar, "Machine learning from theory to algorithms: an overview," in *Journal of physics: conference series*, IOP Publishing, vol. 1142, no. 1, pp. 1-15, 2018.

[32] D. Mouheb, R. Albarghash, M. F. Mowakeh, Z. Al Aghbari, and I. Kamel, "Detection of Arabic cyberbullying on social networks using machine learning," in *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pp. 1-5, Nov 2019.

[33] W. N. H. W. Ali, M. Mohd, and F. Fauzi, "Cyberbullying Predictive Model: Implementation of Machine Learning Approach," in *2021 Fifth International Conference on Information Retrieval and Knowledge Management (CAMP)* IEEE, pp. 65-69, Jun 2021.

[34] B. Haidar, M. Chamoun, and A. Serhrouchni, "A multilingual system for cyberbullying detection: Arabic content detection using machine learning," *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 6, pp. 275-284, Dec 2017.

[35] Y. Liu, Y. Wang, and J. Zhang, "New machine learning algorithm: Random forest," in *International Conference on Information Computing and Applications*, Springer, vol. 7473, pp. 246-252, Sep 2012.

[36] S. K. Kiangala and Z. Wang, "An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment," *Machine Learning with Applications*, vol. 4, pp.1-15, Jun 2021.

[37] T. Kanan, A. Aldaaja, and B. Hawashin, "Cyber-bullying and cyber-harassment detection using supervised machine learning techniques in Arabic social media contents," *Journal of Internet Technology*, vol. 21, no. 5, pp. 1409-1421, Sep 2020.

[38] F. Ouallouche, M. Lazri, and S. Ameur, "Improvement of rainfall estimation from MSG data using Random Forests classification and regression," Atmospheric Research, vol. 211, pp. 62-72, Oct 2018.

[39] E. K. Sahin, "Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest," SN Applied Sciences, vol. 2, no. 7, pp. 1-17, Jun 2020.

[40] Y. Wang et al., "Short-term load forecasting of industrial customers based on SVMD and XGBoost," International Journal of Electrical Power & Energy Systems, vol. 129, pp. 106830, 2021.

[41] I. L. Cherif and A. Kortebi, "On using extreme gradient boosting (XGBoost) machine learning algorithm for home network traffic classification," in *2019 Wireless Days (WD)* IEEE, pp. 1-6, Apr 2019.

[42] R. F. Najeeb and B. N. Dhannoon, "Improving Detection Rate of the Network Intrusion Detection System Based on Wrapper Feature Selection Approach," Iraqi Journal of Science, vol. 59, no.1B, pp. 426-433, Mar 2018.

[43] Hussien, Z. Khalaf, Dhannoon, B. N., "Anomaly Detection Approach Based on Deep Neural Network and Dropout," Baghdad Science Journal, vol. 17, no. (2 Special Issue) NICST, pp. 701-709, 2020.

[44] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295-316, Oct 2020.

[45] M. Feurer and F. Hutter, " Hyperparameter optimization," in *Automated machine learning*: Springer, Cham, pp. 3-33, 2019.

[46] V. V. G. Pentapalli and R. K. Varma, "Cuckoo search optimization and its applications: A review," *Int. J. Adv. Res. Comput. Commun. Eng*, vol. 5, no. 11, pp. 2319-5940, Nov 2016.

[47] L.A.M. Pereira, D. Rodrigues, T.N.S. Almeida, C.C.O. Ramos, A.N. Souza, X.S. Yang, and J.P Papa., "A binary cuckoo search and its application for feature selection," in *Cuckoo search and firefly algorithm*: Springer, vol. 516, pp. 141-154, Jan 2014.

[48] J. Chen, F. Zhao, Y. Sun, and Y. Yin, "Improved XGBoost model based on genetic algorithm," *International Journal of Computer Applications in Technology*, vol. 62, no. 3, pp. 240-245, Apr 2020.