Neural Computing and Applications (2024) 36:1049–1065
https://doi.org/10.1007/s00521-023-09084-w

REVIEW

The effect of rebalancing techniques on the classification performance in cyberbullying datasets

Marwa Khairy[1] · Tarek M. Mahmoud[2] · Tarek Abd-El-Hafeez[3,4]

Received: 21 May 2023 / Accepted: 20 September 2023 / Published online: 6 November 2023
© The Author(s) 2023

Abstract
Cyberbullying detection systems rely increasingly on machine learning techniques. However, class imbalance in cyberbullying datasets, where the percentage of normal labeled classes is higher than that of abnormal labeled ones, presents a significant challenge for classification algorithms. This issue is particularly problematic in two-class datasets, where conventional machine learning methods tend to perform poorly on minority class samples due to the influence of the majority class. To address this problem, researchers have proposed various oversampling and undersampling techniques. In this paper, we investigate the effectiveness of such techniques in addressing class imbalance in cyberbullying datasets. We conduct an experimental study that involves a preprocessing step to enhance machine learning algorithm performance. We then examine the impact of imbalanced data on classification performance for four cyberbullying datasets. To study the classification performance on balanced cyberbullying datasets, we employ four resampling techniques, namely random undersampling, random oversampling, SMOTE, and SMOTE + TOMEK. We evaluate the impact of each rebalancing technique on classification performance using eight well-known classification algorithms. Our findings demonstrate that the performance of resampling techniques depends on the dataset size, imbalance ratio, and classifier used. The conducted experiments proved that there are no techniques that will always perform better the others.

Keywords Classification · Cyberbullying · Undersampling · Oversampling · SMOTE · TOMEK

1 Introduction

In research areas such as machine learning, pattern recognition, and data mining, class imbalance is one of the problems that have recently gained the most attention [1]. Dataset with an unequal class distribution is technically imbalanced; the minority class is represented by a very small number of instances in contrast to the other the majority class.

It is known that cyberbullying could have a negative impact on the people's life in many ways. Machine learning can be effective in detecting the bullies' language patterns and can also create a model for cyberbullying actions to be detected [2]. It is difficult to conduct some machine learning research in the field of cyberbullying, not because of the lack of accurately labeled datasets, but also because all available datasets suffer from a class imbalance where the majority (not bullying) class usually greatly outnumbers the minority (bullying) class [3, 4]. In cyberbullying datasets, the percentage of normal labeled classes is higher than the percentage of abnormal labeled ones, which is called as class imbalance problem in data mining. If training dataset is imbalanced, the classification algorithm generally predicts the labels of the majority class instances correctly and

✉ Marwa Khairy
Marwa.kh.mohamed@mu.edu.eg

✉ Tarek Abd-El-Hafeez
tarek@mu.edu.eg

Tarek M. Mahmoud
tarek@fcai.usc.edu.eg

[1] Faculty of Computers and Information, Minia University, EL-Minia, Egypt

[2] Faculty of Computers and Artificial Intelligence, Computer Science Department, University of Sadat City, Sadat City, Egypt

[3] Computer Science Department, Faculty of Science, Minia University, EL-Minia, Egypt

[4] Deraya University, EL-Minia, Egypt

the minority class instances incorrectly which leads to a major problem for cyberbullying detection systems [5, 6].

Class imbalance is a persistent challenge in data mining, particularly in the context of cyberbullying detection systems. Resampling techniques have been proposed as a potential solution to this issue, with data preprocessing considered essential for building effective models using modern data mining algorithms. Among the simplest sampling methods are random undersampling and random oversampling. The former involves selecting random samples from the majority class to be deleted, while the latter randomly duplicates minority class samples. However, these techniques are not without their limitations, as undersampling can result in information loss, while oversampling can lead to over-fitting. To overcome these issues, various alternative techniques have been proposed.

Nitesh V. Chawla et al. [7] have developed an oversampling technique called synthetic minority oversampling technique (SMOTE). The experiments of this technique were performed using C4.5, Ripper, and naive Bayes classifier algorithms. The obtained results of applying this approach showed that the accuracy of classifiers of minority class is improved. Because of this success, the algorithm has been used in many areas of data mining. The minority class in the datasets affects the classification accuracy of classification algorithms. A clustering-based undersampling technique was developed by Yen et al. [8] to enhance the classification accuracy for minority class. Experimental results demonstrate that other undersampling techniques are outperformed by clustering-based undersampling techniques. Classification methods developed by researchers are used in many important areas. To increase the classification accuracy of medical datasets, Li et al. [9] used oversampling and undersampling strategies. Liu et al. [10] tested that the oversampling and undersampling techniques on the imbalanced text dataset affected the performance and classification accuracy. In this study, we investigate the effect of four resampling techniques in the performance of four cyberbullying datasets.

The paper is organized as follows: Sect. 2 presents the related work, providing a comprehensive overview of the existing literature. In Sect. 3, the background is provided, covering resampling techniques, classification algorithms, and performance measures relevant to the study. The methodology employed in this research is detailed in Sect. 4. Section 5 presents the results obtained from the experiments and discusses the key findings. A thorough discussion of the results is provided in Sect. 6, while Sect. 7 concludes the paper with final remarks, limitations, and future work.

2 Related work

In this section, we look at few studies that used machine learning methods and resampling strategies to solve the problem of an imbalanced dataset.

Kubȷt and Matwin [10] suggested removing noisy and duplicated training data by using a one-sided selection method that decreases the majority class. A SMOTE algorithm was proposed by Chawla et al. [7] to increase minority classes. The benefit of using SMOTE is reducing over-fitting problem because synthetic examples are made instead of replicating instances, and this is caused by random oversampling. Furthermore, there is no loss of important information; thus, the decision areas become broader and less specialized.

Naseriparsa et al. [11] suggest a new hybrid approach in which a combination of resampling, sample domain filtering, and genetic search wrapper subset evaluation method has been used to minimize the Lung-Cancer dataset dimensions derived from the UCI Machine Learning Repository databases. Khaldy and Kambhampati [12] highlighted the challenge of implementing five alternative feature selections and illustrated the usefulness of imbalanced class for the high-dimensional dataset. In medical datasets, Mehmet and Mohammed [13] looked at the effects of oversampling and undersampling techniques. Several medical benchmark datasets and well-known classification techniques are used. Experimental results show that oversampled datasets can learn more efficiently and predict patient instances more successfully.

Regarding the study of imbalance problem in a cyberbullying case, Colton and Hofmann [4] examine the performance of a prediction model whether it is affected by using resampling strategies or not. A compromise method is also investigated, in which the positive class is partially oversampled and the negative class is partially undersampled. Although sampling using the most often seen features was not exactly a class imbalance solution, it was investigated.

Talpur and O'Sullivan [6, 14] have recently addressed the issue of class imbalance in cyberbullying datasets. For their research, they used the SMOTE oversampling approach. The results revealed that when the SMOTE parameter was enabled, the base classifier's overall performance improved marginally as it dealt with the distribution of class imbalance.

Table 1 presents an overview of various rebalancing techniques along with their advantages, disadvantages, cost considerations, and impact on classification performance. Six different techniques are discussed, including random oversampling, random undersampling, SMOTE, ADASYN, TOMEK Links, and cost-sensitive learning.

Table 1 An overview of various rebalancing techniques

Rebalancing Technique	Description	Advantages	Disadvantages	Cost Considerations	Impact on Classification Performance
1. Random oversampling [15]	Duplicates instances from the minority class randomly until class balance is achieved	Simple to implement Can improve recall for the minority class	May lead to over-fitting No new information is added, which may result in model bias	Low computational cost as it involves replicating existing data. No additional data acquisition required	Moderate improvement in the minority class recall. Increased accuracy on the minority class, but potential over-fitting may impact overall performance on the test set
2. Random undersampling [16]	Removes instances from the majority class randomly until class balance is achieved	Simple to implement Reduces computation time Can improve training time for some algorithms	May discard valuable information, leading to underfitting May not be effective for severe imbalances	Low computational cost as it involves removing data from the majority class Reduced memory requirements	Improved training time, but reduced overall accuracy due to data loss
3. SMOTE (synthetic minority oversampling technique) [17]	Creates synthetic instances for the minority class using interpolation	Addresses over-fitting by generating new information Effective at improving performance for the minority class	May introduce noisy samples Can lead to model overgeneralization	Moderate computational cost as it requires generating synthetic data points No additional data acquisition required	Significant improvement in the minority class recall and overall accuracy. Potential reduction in precision due to the introduction of synthetic samples
4. ADASYN (adaptive synthetic sampling) [18]	Similar to SMOTE but introduces more synthetic instances near the decision boundary	Focuses on more challenging samples, improving the decision boundary Better suited for severe imbalances	May introduce more noise than SMOTE. Computationally more expensive than SMOTE	Moderate computational cost as it generates synthetic data points near the decision boundary No additional data acquisition required	Enhanced performance on challenging samples near the decision boundary. Improved accuracy in the minority class, but the additional noise may impact precision and overall performance
5. TOMEK Links [19]	Identifies pairs of instances from different classes that are nearest neighbors and removes the majority class instance	Can improve the decision boundary between classes	May not be effective for high-dimensional data	Low computational cost as it involves identifying and removing instances No additional data acquisition required	Better decision boundary, but minimal improvement in overall accuracy. May result in slightly reduced dataset size
6. Cost-sensitive learning [20]	Adjusts the classification algorithm's cost matrix to reflect the imbalance	Tailored solution for imbalanced datasets Can be used with various classification algorithms	Requires setting the right cost matrix, which may be challenging Performance depends on the chosen cost matrix	Minimal computational cost during model training Potential additional cost to collect class misclassification information and evaluation	Improves the overall classification performance and addresses the imbalance. Increased accuracy on the minority class and improved F1-score due to a stronger focus on the minority class during training. However, if the cost matrix is not well defined, it may lead to unintended consequences, such as prioritizing the majority class

Table 2 Cyberbullying datasets

Study	Platform	Language	Size	Balancing
[21]	Formspring	English	3915	0.142
[22]	YouTube Formspring	English	–	–
[23]	Twitter, MySpace	English	1,570,000	–
[24]	YouTube	English	4626	0.097
[25]	Twitter	English	4865	0.019
[26]	Kaggle	English	2647	0.272
[27]	Twitter	English	1340	0.152
[28]	Ask FM	Dutch	85,485	0.067
[29]	Schoolboard Bulletins (BBS)	Japanese	2222	0.128
[30]	Twitter	English	4865	0.186
[31]	Twitter	English	10,007	0.06
[32]	Twitter	English	1762	0.388
[33]	Train-Formspring and MySpace Test-Twitter	English	3279	0.12
[34]	Instagram	English	1954	0.29
[35]	Formspring	English	13,000	0.066
[36]	Formspring	English	13,160	0.194

Table 3 The previous work done in Arabic cyberbullying detection

Study	Dataset			Feature Representation	Classifier	Performance			
	Plat form	Size	Classes			Acc	P	R	F
[37]	Twitter	Arabic=35,273 English=91,431	Yes/no	Tweet to SentiStrength Feature Vector	Naïve Bayes SVM		93.4	94.1	92.7
[38]	Twitter	Large=34,890, small=4913	Yes/no	Word embedding	FFNN	94.5			
[39]	Twitter	34,890	Bully/non-bully		Bagging, boosting (KNN, SVM,NB)		93.3	93.5	92.0
[40]	Twitter	Real-time classification		TF-IDF					
[41]	YouTube and Twitter	25,000		TF-IDF	Naive Bayes (NB)	95.9	92.9	92.5	92.7
[42]	YouTube Twitter	training (100,327), testing (2020)		TF-IDF	PMI, Chi-square entropy	81.0, 62.1, 39.1			
[43]	Aljazeera.net. (test) Twitter	32K	CB, NCB	Word embedding TF-IDF, n-gram, bow	CNN, RNN				84.0
[44]	Facebook and Twitter	6138	Positive/ negative	TF-IDF	KNN, SVM, NB, random forest, and J48		94.5	94.4	94.4
[45]	Twitter		Bullying/ no bullying	Sentiment analysis, the emojis, and user history		85.0			
[46]	Twitter	151,000		Sentiment analysis	Ridge regression (RR) and logistic regression (LR)				

Table 2 lists several datasets related to cyberbullying, along with some key information about each dataset. The first column lists the study or source of the dataset. The second column specifies the platform where the data was

Table 4 Advantages and disadvantages of used resampling techniques

	Advantages	Disadvantage
Random undersampling	When the training dataset is large, it can help with run time and storage issues	It eliminates potentially useful. As a result, the actual test dataset gave inaccurate results
Random oversampling	No loss of useful information	It may cause over-fitting as it replicates cases of minority class
SMOTE	No loss of useful information and no over-fitting via generating synthetic examples	Increase overlapping of classes and produce more noise
SMOTE + TOMEK	Remove noisy points from both classes and better classifier efficiency fitting on the transformed dataset	Eliminating some useful features

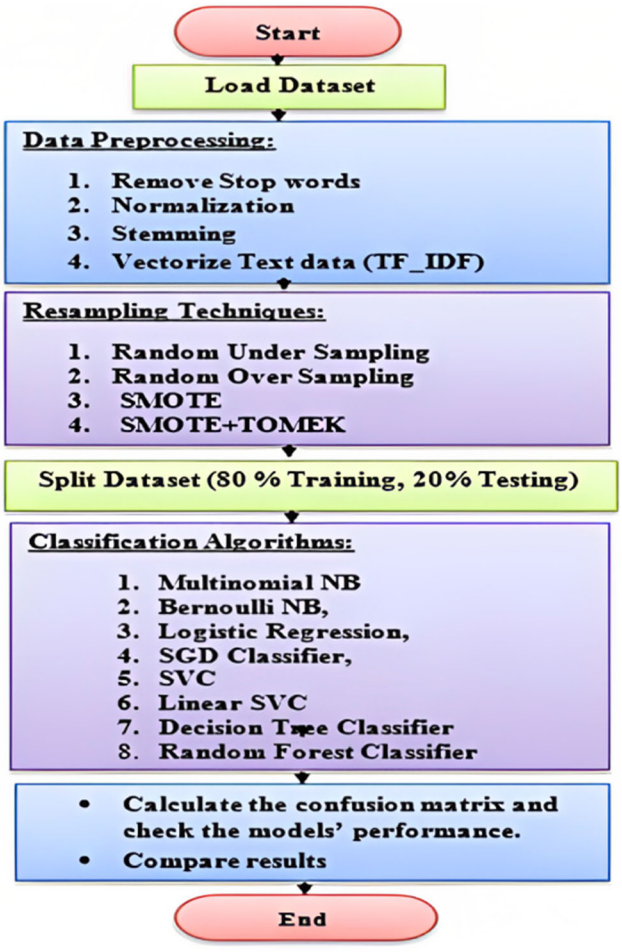

Fig. 1 Proposed method for this study

collected from, such as Formspring, Twitter, Instagram, and MySpace. The third column indicates the language of the data, which is mostly in English but also includes Dutch and Japanese. The fourth column shows the size of each dataset, which ranges from 1340 to 1,570,000 instances. Some

Table 5 Cyberbullying dataset used

Dataset	Size	Class 0	Class 1	Imbalanced Proportion
Kaggle	8005	5398	2607	2.07: 1
Twitter1	8316	5948	2368	2.51: 1
Twitter2	10,971	7595	3376	2.25: 1
YouTube	2745	2944	451	6.53: 1

datasets are balanced, meaning that the proportion of positive and negative instances is roughly equal, while others are imbalanced, with a higher proportion of one class compared to the other.

The fifth column provides information about the balancing of each dataset, specified as a decimal value between 0 and 1. For example, a value of 0.142 in the balancing column means that 14.2% of instances in the dataset belong to the positive class, while the remaining 85.8% belong to the negative class. Table 2 provides useful information for researchers interested in studying cyberbullying and developing machine learning models to detect and prevent it.

Table 3 specifically focuses on previous work done in Arabic cyberbullying detection. The first column lists the study or source of the dataset. The second column specifies the dataset used for the study, which includes Twitter in Arabic and English and Aljazeera.net. The third column indicates the feature representation used in the study, which includes SentiStrength Feature Vector, word embeddings, TF-IDF, and n-gram. The fourth column lists the classifier used in each study, which includes naive Bayes, SVM, KNN, random forest, logistic regression, and convolutional neural networks (CNN) and recurrent neural networks (RNN). The fifth column provides the performance metrics of each classifier, such as accuracy (Acc), precision (P), recall (R), and F1 score (F), which are commonly used

Table 6 Performance evaluation of classifiers with resampling techniques for the Kaggle dataset

Technique	Algorithm	Acc	F1	Recall	Precision
Kaggle dataset					
Unbalanced	Multinomial NB	0.775	0.754	0.775	0.780
	Bernoulli NB	0.780	0.780	0.780	0.780
	Logistic regression	0.792	0.774	0.792	0.799
	SGD	0.788	0.782	0.788	0.783
	SVC	0.792	0.775	0.792	0.798
	Linear SVC	0.779	0.773	0.779	0.774
	Decision tree	0.731	0.732	0.731	0.733
	Random forest	0.778	0.772	0.778	0.772
Random undersample	Multinomial NB	0.779	0.778	0.779	0.782
	Bernoulli NB	0.651	0.614	0.614	0.734
	Logistic regression	0.762	0.761	0.762	0.763
	SGD	0.790	0.789	0.790	0.791
	SVC	0.763	0.762	0.763	0.764
	Linear SVC	0.786	0.785	0.786	0.787
	Decision tree	0.697	0.697	0.697	0.698
	Random forest	0.747	0.747	0.747	0.749
Random oversample	Multinomial NB	0.868	0.866	0.868	0.878
	Bernoulli NB	0.692	0.659	0.692	0.790
	Logistic regression	0.847	0.847	0.847	0.848
	SGD	0.891	0.891	0.891	0.894
	SVC	0.911	0.911	0.911	0.911
	Linear SVC	0.897	0.897	0.897	0.899
	Decision tree	0.853	0.852	0.853	0.859
	Random forest	0.916	0.916	0.916	0.916
SMOTE	Multinomial NB	0.757	0.761	0.757	0.768
	Bernoulli NB	0.695	0.703	0.695	0.745
	Logistic regression	0.750	0.753	0.750	0.760
	SGD	0.736	0.739	0.736	0.743
	SVC	0.778	0.769	0.778	0.772
	Linear SVC	0.727	0.732	0.727	0.740
	Decision tree	0.719	0.721	0.719	0.724
	Random forest	0.768	0.770	0.768	0.772
SMOTE + TOMEK	Multinomial NB	0.761	0.765	0.761	0.772
	Bernoulli NB	0.697	0.705	0.697	0.746
	Logistic regression	0.762	0.765	0.762	0.771
	SGD	0.740	0.744	0.740	0.753
	SVC	0.784	0.774	0.784	0.780
	Linear SVC	0.731	0.735	0.731	0.744
	Decision tree	0.729	0.730	0.729	0.732
	Random forest	0.757	0.758	0.757	0.760

in machine learning to evaluate the quality of a classifier's predictions. Table 3 provides an overview of the different approaches used in Arabic cyberbullying detection studies and their corresponding performance metrics, which can be useful for researchers working in the field.

3 Resampling techniques

This study aims to investigate the impact of four resampling techniques, namely random undersampling, random oversampling, SMOTE, and hybrid (SMOTE and TOMEK Links), on unbalanced cyberbullying datasets, and these techniques can be summarized as follows:

Table 7 Comparison between recall and precision before and after SMOTE

SVC	None		SMOTE	
	Recall	Precision	Recall	Precision
Class0	0.95	0.78	0.90	**0.80**
Class1	0.49	0.83	**0.55**	0.73

1. SMOTE (Synthetic minority oversampling technique)

   SMOTE is a popular oversampling technique used to address class imbalance in datasets. Class imbalance occurs when one class has significantly fewer instances than the other, leading to biased learning algorithms. SMOTE helps alleviate this issue by generating synthetic examples for the minority class, thereby balancing the class distribution.

   The basic idea behind SMOTE is to create synthetic instances by interpolating between existing minority class instances. Here is how it works:

- For each minority class instance, SMOTE selects its k-nearest neighbors in the feature space.
- Synthetic instances are generated by randomly selecting one of the k neighbors and creating a new instance by interpolating between the selected neighbor and the original instance.
- This process is repeated until the desired level of oversampling is achieved.

   SMOTE effectively increases the number of minority class instances, making the dataset more balanced and improving the performance of learning algorithms. However, it does not address potential overlapping or noisy samples that might exist in the dataset.

2. SMOTE + TOMEK Links:

   SMOTE+TOMEK Links is a hybrid resampling technique that combines the SMOTE oversampling method with the TOMEK Links undersampling technique. The goal of this combination is to not only increase the number of minority class instances but also remove potential noisy samples and enhance the separation between different classes. TOMEK Links are pairs of instances from different classes that are close to each other but considered to be ambiguous or noisy. By removing these instances, TOMEK Links aim to improve the decision boundary between classes. Here is how SMOTE+TOMEK Links works:

- First, SMOTE is applied to oversample the minority class and generate synthetic instances.
- Next, TOMEK Links are used to identify pairs of instances with different class labels that are close to each other.
- For each identified TOMEK Link, the instance from the majority class is removed.
- The resulting dataset consists of the augmented minority class instances and the remaining majority class instances.

By combining SMOTE and TOMEK Links, this approach helps to both increase the representation of the minority class and address potential noisy samples, resulting in a more balanced and better separated dataset. This, in turn, can lead to improved classification performance and more reliable predictions. SMOTE and SMOTE + TOMEK Links are valuable techniques for handling class imbalance in datasets, and they have proven to be effective in various machine learning applications.

3.1 Undersampling techniques

By randomly deleting examples of the majority class, undersampling techniques attempt to balance class distribution. This is repeated until the dominant and minority classes' situations are equalized. Some of the undersampling approaches that are more commonly used and applied include random undersampling (RUS) and TOMEK Links undersampling [47].

3.2 Oversampling techniques

Oversampling methods either replicate or create new instances in the minority class. Oversampling strategies includes many techniques such as random oversampling (ROS) and synthetic minority oversampling technique (SMOTE) [47].

3.3 Hybrid techniques

While oversampling or undersampling techniques can be effective when applied individually to a training dataset, a combination of both techniques can yield a model that better fits the overall results on the transformed dataset. SMOTE is the most popular and widely used oversampling technique and is often paired with one of several undersampling techniques. The following are some of the frequently used and implemented combinations of data sampling methods [47]

Table 8 Performance evaluation of classifiers with resampling techniques for the Twitter1 dataset

Technique	Algorithm	Acc	F1	Recall	Precision
Twitter1					
Unbalanced	Multinomial NB	0.719	0.623	0.719	0.671
	Bernoulli NB	0.693	0.660	0.693	0.651
	Logistic regression	0.724	0.655	0.724	0.684
	SGD	0.714	0.650	0.714	0.662
	SVC	0.732	0.650	0.732	0.721
	Linear SVC	0.695	0.662	0.695	0.653
	Decision tree	0.622	0.616	0.622	0.612
	Random forest	0.709	0.651	0.709	0.655
Random undersample	Multinomial NB	0.575	0.573	0.575	0.578
	Bernoulli NB	0.568	0.545	0.568	0.583
	Logistic regression	0.578	0.577	0.578	0.579
	SGD	0.591	0.591	0.591	0.591
	SVC	0.597	0.592	0.597	0.602
	Linear SVC	0.586	0.586	0.586	0.586
	Decision tree	0.554	0.554	0.554	0.554
	Random forest	0.595	0.595	0.595	0.597
Random oversample	Multinomial NB	0.823	0.821	0.823	0.833
	Bernoulli NB	0.784	0.737	0.784	0.793
	Logistic regression	0.789	0.789	0.789	0.790
	SGD	0.839	0.838	0.839	0.842
	SVC	0.909	0.909	0.909	0.910
	Linear SVC	0.847	0.846	0.847	0.853
	Decision tree	0.785	0.781	0.785	0.807
	Random forest	0.884	0.883	0.884	0.884
SMOTE	Multinomial NB	0.612	0.629	0.612	0.661
	Bernoulli NB	0.617	0.629	0.617	0.684
	Logistic regression	0.637	0.648	0.637	0.667
	SGD	0.635	0.646	0.635	0.661
	SVC	0.717	0.669	0.717	0.675
	Linear SVC	0.608	0.623	0.608	0.649
	Decision tree	0.626	0.628	0.626	0.631
	Random forest	0.683	0.671	0.683	0.663
SMOTE + TOMEK	Multinomial NB	0.620	0.635	0.620	0.662
	Bernoulli NB	0.626	0.633	0.626	0.643
	Logistic regression	0.646	0.656	0.646	0.673
	SGD	0.626	0.641	0.626	0.670
	SVC	0.717	0.671	0.717	0.675
	Linear SVC	0.613	0.629	0.613	0.658
	Decision tree	0.621	0.625	0.621	0.630
	Random forest	0.681	0.668	0.681	0.660

- The most basic combination is SMOTE with random undersampling, which has been shown to outperform SMOTE alone.
- SMOTE with TOMEK Links and SMOTE with Edited Nearest Neighbors Rule are used to remove noisy points from both classes at the class boundary, which appears to improve classifier performance on the altered data.

Table 4 illustrates the advantages and disadvantages between the four resampling that we use in this study [48].

Table 9 Comparison between recall and precision before and after SMOTE

SVC	None		SMOTE	
	Recall	Precision	Recall	Precision
Class0	0.98	0.73	93	75
Class1	0.09	0.69	19	50

4 Methods

Figure 1 depicts the methodology employed in this study. The process includes dataset selection, data preprocessing, application of machine learning classifiers, and evaluation of performance measures. Details on each of these steps are provided in the following.

4.1 Datasets

This study is implemented using four imbalanced cyberbullying datasets. These datasets have been found publicly at [49], and they contain different sizes and different imbalance ratios. Table 5 gives the original distribution of the data in terms of the source, size of the dataset, number of their majority and minority instances, and their imbalance ratio (IR).

4.2 Dataset preparation

Preprocessing and resampling techniques play a crucial role in data analysis and machine learning tasks. These techniques are employed to preprocess and manipulate the data before feeding it into a learning algorithm, with the aim of improving the quality and reliability of the results. Let us explore these techniques in more detail:

1. Preprocessing techniques:
   Preprocessing involves a series of steps to transform and prepare the data for analysis. Some common preprocessing techniques include:
   - *Data Cleaning* This involves handling missing data, removing outliers, and dealing with inconsistent or erroneous values. It ensures the data are accurate and reliable.
   - *Feature Scaling* It is important to scale features to a consistent range to prevent certain features from dominating the learning process. Common scaling methods include standardization (mean of 0 and variance of 1) and normalization (scaling to a specified range).
   - *Feature Encoding* Categorical variables often need to be converted into numerical representations for machine learning algorithms to process. Techniques like one-hot encoding and label encoding are commonly used for this purpose.
   - *Dimensionality Reduction* When dealing with high-dimensional data, dimensionality reduction techniques like principal component analysis (PCA) or feature selection methods can be applied to reduce the number of features while retaining important information.

2. Resampling techniques:
   Resampling techniques are used to address class imbalance issues in the dataset, where the number of instances in one class significantly outweighs the number in another class. Some commonly used resampling techniques include:
   - *Oversampling* This involves increasing the number of instances in the minority class by duplicating or generating synthetic samples. Techniques like SMOTE (synthetic minority oversampling technique) generate synthetic examples by interpolating between existing minority class instances.
   - *Undersampling* This technique aims to reduce the number of instances in the majority class by randomly selecting a subset of instances. Undersampling can be effective when the majority class has a large number of redundant or similar instances.
   - *Hybrid Approaches* These techniques combine oversampling and undersampling to achieve a more balanced dataset. For example, one popular approach is SMOTE combined with TOMEK Links, where synthetic samples are generated for the minority class, and TOMEK Links are used to remove noisy samples from both classes.

These preprocessing and resampling techniques are essential for preparing data for effective machine learning model training. By properly handling data cleaning, scaling, encoding, and addressing class imbalance, these techniques contribute to more accurate and reliable predictions, ultimately enhancing the performance of machine learning models.

In data preparation, standard preprocessing steps are implemented. The following steps are performed for each dataset:

1. Remove stop words.
2. Normalization.

Table 10 Performance evaluation of classifiers with resampling techniques for the Twitter dataset

Technique	Algorithm	Acc	F1	Recall	Precision
Twitter					
Unbalanced	Multinomial NB	0.815	0.807	0.815	0.810
	Bernoulli NB	0.825	0.823	0.825	0.822
	Logistic regression	0.825	0.814	0.825	0.820
	SGD	0.829	0.820	0.829	0.825
	SVC	0.830	0.820	0.830	0.828
	Linear SVC	0.827	0.822	0.827	0.822
	Decision tree	0.798	0.798	0.798	0.798
	Random forest	0.833	0.828	0.833	0.829
Random undersample	Multinomial NB	0.779	0.777	0.779	0.794
	Bernoulli NB	0.773	0.767	0.773	0.798
	Logistic regression	0.775	0.775	0.775	0.776
	SGD	0.810	0.810	0.810	0.812
	SVC	0.783	0.783	0.783	0.783
	Linear SVC	0.811	0.811	0.811	0.813
	Decision tree	0.739	0.739	0.739	0.739
	Random forest	0.776	0.772	0.776	0.793
Random oversample	Multinomial NB	0.860	0.858	0.860	0.875
	Bernoulli NB	0.892	0.892	0.892	0.894
	Logistic regression	0.873	0.873	0.873	0.873
	SGD	0.909	0.909	0.909	0.910
	SVC	0.929	0.929	0.929	0.929
	Linear SVC	0.920	0.920	0.920	0.922
	Decision tree	0.882	0.882	0.882	0.886
	Random forest	0.936	0.936	0.936	0.937
SMOTE	Multinomial NB	0.791	0.797	0.791	0.811
	Bernoulli NB	0.817	0.818	0.817	0.820
	Logistic regression	0.790	0.795	0.790	0.806
	SGD	0.766	0.773	0.766	0.795
	SVC	0.821	0.816	0.821	0.816
	Linear SVC	0.770	0.776	0.770	0.789
	Decision tree	0.789	0.791	0.789	0.792
	Random forest	0.823	0.822	0.823	0.821
SMOTE+TOMEK	Multinomial NB	0.787	0.793	0.787	0.807
	Bernoulli NB	0.814	0.816	0.814	0.818
	Logistic regression	0.788	0.793	0.788	0.804
	SGD	0.779	0.784	0.779	0.795
	SVC	0.817	0.812	0.817	0.812
	Linear SVC	0.784	0.789	0.784	0.800
	Decision tree	0.787	0.787	0.787	0.787
	Random forest	0.826	0.825	0.826	0.825

3. Stemming.
4. Transform text data to numerical via vectorizing it and calculate the (TF_IDF).

4.3 Used resampling techniques

To investigate the impact of resampling techniques, the four methods (random undersampling, random oversampling, SMOTE, and SMOTE+TOMEK) are applied to each dataset, resulting in a balanced dataset that is then used in the classification phase.

Table 11 Comparison between recall and precision before and after SMOTE

Linear SVC	None		SMOTE	
	Recall	Precision	Recall	Precision
Class0	0.91	0.85	0.79	0.87
Class1	0.63	0.75	0.73	0.59

4.4 Machine learning classification algorithms

After resampling the dataset, the dataset was split as 80 % for training and 20 % for testing. Then, it was passed to the classification phase in which eight machine learning classifiers were used (multinomial NB, Bernoulli NB, logistic regression, SGD classifier, SVC, linear SVC, decision tree classifier, and random forest classifier). The choice of optimizer depends on factors such as the problem, dataset characteristics, and training requirements. SGD (stochastic gradient descent) is often chosen for its computational efficiency, scalability to large datasets, noise tolerance for better generalization, flexibility in hyperparameter tuning, and suitability for online learning. However, it has limitations and may require careful tuning. Experimentation with different optimizers is recommended to find the optimal choice, considering factors such as network architecture, dataset, and computational resources.

4.5 Performance measures

Simpler measurements, such as accuracy score, can be misleading. As a result, we calculate the confusion matrix and use the accuracy, precision, recall, and F1_score metrics for each classifier to assess its performance.

5 Experimental results

This section report results from a selection of experiments on the classification of cyberbullying datasets under different scenarios of resampling. All the experimental analyses were implemented using the Python library imbalanced-learn. It is compatible with Scikit-learn. The Scikit-learn is a machine learning module that provides simple and efficient tools for data mining and machine learning. The machine learning classification algorithms used in our investigation are multinomial NB, Bernoulli NB, logistic regression, SGD classifier, SVC, linear SVC, decision tree classifier, and random forest classifier**.** The performances of the eight algorithms for the four datasets are assessed and compared. In order to identify the best classification algorithm and the best resampling technique, the algorithms are compared with to their performance. The values obtained for each dataset are shown in the following tables. Hulse et al. [50] suggest that the utility of the resampling methods depends on several factors, including the ratio between positive and negative examples, other characteristics of data, and the nature of the classifier.

Table 6 illustrates the effect of the used resampling techniques on the performance measures (accuracy, F1 score, recall, and precision) for all classifiers in the first dataset (Kaggle). First for the unbalance dataset, we can see that logistic regression and SVC outperform the other classifiers for all performance measures except F1 score the SGD classifier is the higher. Logistic regression can be competitive in the case of highly unbalanced data [51].

There are large amounts of data discarded in random undersampling. This can be extremely troublesome, as the lack of such data can make it more difficult to learn the decision boundary between minority and majority instances, resulting in a loss of classification results [51, 52]. From Table 3, we notice a decrease in all classifier's performance except the multinomial NB and the SGD classifier which achieve the best performance.

For all classifiers, ROS technique has higher for all performance measures values than other methods. Random forest gets the highest performance between all classifiers. This technique can affect models that seek good splits of the data, such as support vector machines and decision trees [53]. Comparing decision tree performance in this technique with the original and undersample, we notice an improvement in its performance. The main drawback with oversampling is increasing the likelihood of over-fitting since it duplicates the minority class events. A second drawback is increasing the learning time as it increases the number of training examples.

The problem of over-fitting caused by random oversampling is prevented in SMOTE as synthetic examples rather than replication of instances are created. Also, there is no loss of useful information. After using SMOTE we notice that the recall on the minority class increased while maintaining a high precision on the majority class which is desired by classification algorithms. For example, Table 7 illustrates the SVC classification report before SMOTE and after. One can notice that the recall for the minority class1 is increased, while the precision for the majority class0 is increased. Random forest and SVC outperform the other classifiers for all measures.

SMOTE does not take neighboring examples from other groups into consideration when creating synthetic examples. This can lead to increased class overlap and additional noise can be added, so the hybrid (SMOTE with TOMEK) was used. The results showed an improvement occurred to all performance measures than using SMOTE alone for

Table 12 Performance evaluation of classifiers with resampling techniques for the YouTube dataset

Technique	Algorithm	Acc	F1	Recall	Precision
YouTube					
Unbalanced	Multinomial NB	0.881	0.827	0.881	0.779
	Bernoulli NB	0.735	0.773	0.735	0.835
	Logistic regression	0.883	0.831	0.883	0.839
	SGD	0.857	0.845	0.857	0.836
	SVC	0.883	0.828	0.883	0.779
	Linear SVC	0.869	0.842	0.869	0.829
	Decision tree	0.789	0.797	0.789	0.805
	Random forest	0.883	0.831	0.883	0.839
Random undersample	Multinomial NB	0.580	0.508	0.580	0.619
	Bernoulli NB	0.530	0.464	0.530	0.622
	Logistic regression	0.596	0.594	0.596	0.594
	SGD	0.596	0.591	0.596	0.596
	SVC	0.585	0. 585	0.585	0. 585
	Linear SVC	0.613	0.611	0.613	0.611
	Decision tree	0.524	0.525	0.524	0.525
	Random forest	0.558	0.558	0.558	0.561
Random oversample	Multinomial NB	0.598	0.517	0.598	0.766
	Bernoulli NB	0.823	0.820	0.823	0.849
	Logistic regression	0.905	0.905	0.905	0.913
	SGD	0.981	0.981	0.981	0.981
	SVC	0.990	0.990	0.990	0.990
	Linear SVC	0.974	0.974	0.974	0.975
	Decision tree	0.867	0.865	0.867	0.891
	Random forest	0.980	0.980	0.980	0.980
SMOTE	Multinomial NB	0.705	0.755	0.705	0.851
	Bernoulli NB	0.755	0.787	0.755	0.834
	Logistic regression	0.747	0.784	0.747	0.847
	SGD	0.760	0.792	0.760	0.844
	SVC	0.877	0.830	0.877	0.810
	Linear SVC	0.741	0.777	0.741	0.835
	Decision tree	0.750	0.779	0.750	0.820
	Random forest	0.862	0.828	0.862	0.807
SMOTE + TOMEK	Multinomial NB	0.696	0.748	0.696	0.850
	Bernoulli NB	0.750	0.782	0.750	0.829
	Logistic regression	0.745	0.783	0.745	0.847
	SGD	0.751	0.786	0.751	0.841
	SVC	0.877	0.830	0.877	0.810
	Linear SVC	0.752	0.786	0.752	0.837
	Decision tree	0.720	0.753	0.720	0.795
	Random forest	0.874	0.841	0.874	0.829

most classifiers as (SMOTE with TOMEK) removes noisy points along the class boundary from both classes, which seems to have the effect of the better performance of classifiers fit on the transformed dataset. The combination was shown to provide a reduction in false negatives at the cost of an increase in false positives for a binary classification task. The hybrid method improved recall and lowered the FN/FP ratio for every classifier, indicating improved sensitivity to cyberbullying [54]. SVC outperforms the other classifiers for all measures.

Table 8 illustrates the performance of resampling techniques for the Twitter 1 dataset. For the unbalance dataset, one can notice that SVC and linear SVC outperforms the other classifiers. For the RUS technique, one can notice a

Table 13 Recall and precision for both classes for all resampling techniques

Linear SVC	None		RUS		ROS		SMOTE		SMOTE + TOMEK	
	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
Class0	0.97	0.89	0.61	0.58	0.96	1.0	0.78	0.92	0.79	0.92
Class1	0.13	0.34	0.62	0.65	1.0	0.96	0.47	0.22	0.48	0.23

decrease in all performance measures for all classifiers. As mentioned before, this is because there are large amounts of data discarded in random undersampling. This can be extremely troublesome, as the lack of such data can make it more difficult to learn the decision boundary between minority and majority instances, resulting in a loss of classification results. [51, 52].

Unlike RUS, the ROS technique leads to no information loss and outperforms RUS. ROS achieves the highest performance than other methods for all performance measures values. SVC gets the highest performance between all classifiers. As mentioned before, the main drawback with oversampling is increasing the likelihood of over-fitting since it duplicates the minority class events. A second drawback is increasing the learning time as it increases the number of training examples. For that, we use the SMOTE technique.

The results of the SMOTE technique showed that it achieved a high recall on the minority class while maintaining a high precision on the majority class, which is desirable for classification algorithms. The classification report for the SVC before and after SMOTE showed an increase in recall for the minority class and an increase in precision for the majority class. Additionally, some classifiers showed an improvement in precision and F1_score compared to the original dataset. The SVC classifier achieved the highest performance among all classifiers (Table 9).

Regarding the hybrid (SMOTE with TOMEK), as in the previous dataset the results showed an improvement occurred to all performance measures than using SMOTE alone for most classifiers as (SMOTE with TOMEK) eliminates noisy points from both classes along the class boundary, which achieves a better performance of classifiers on the transformed dataset. Table10 illustrates the performance of resampling techniques for the Twitter 2 dataset. For the original unbalanced dataset, the size is 10,971 which is larger than the two previous datasets. Our empirical results, consistent with [55, 56], confirm that size of the training set and the classification rate are indeed correlated. Although these algorithms perform relatively well with small datasets, all used classifiers show a major improvement in performance as the number of cases increases, indicating a more consistent learning method. All classifiers work well compared to previous datasets. Random forest achieves the best performance. Regarding RUS results, although the performance measures are decreased than in the original unbalanced dataset, we notice that precision is higher than the recall for most of classifiers. Higher precision means that an algorithm returns more relevant results than irrelevant ones. Linear SVC than SGD achieves the best performance.

As noticed before, ROS achieved the best results between the used sampling techniques. All performance measures for all classifiers increased than for the original and the RUS technique. Random forest is the best classifier.

Regarding the SMOTE technique, as in the previous datasets we notice that a high recall on the minority class while maintaining a high precision on the majority class which is desired by classification algorithms. For example, Table 11 illustrates the linear SVC classification report before SMOTE and after. One can notice that the recall for the minority class1 is increased, while the precision for the majority class0 is increased.

As observed in the previous datasets, the combination of SMOTE with TOMEK Links leads to improved performance in certain classifiers (namely SGD, linear SVC, and random forest) when compared to using SMOTE alone. This can be attributed to the fact that SMOTE with TOMEK Links removes noisy data points from both classes along the class boundary, leading to better classifier efficiency on the transformed dataset. Among all the classifiers used, random forest exhibits the highest performance.

Table 12 illustrates the performance of resampling techniques for the YouTube dataset. For the original unbalanced dataset, one can notice that the size is 2745 less than the previous three datasets and the imbalance ration is 6.53: 1 which is high than previous datasets. Hulse et al. [52] suggest that the utility of the resampling methods depends on a number of factors, including the ratio between positive and negative examples, other characteristics of data, and the nature of the classifier. For the datasets with a severe imbalance, our experiments consistent with [57], one can observe that all the resampling techniques improve the recall of the minority class (TP rate) and the precision of the majority class except for the RUS technique because of

information loss that occurred by reducing training data sample size. See Table 13 (linear SVC as an example).

Table 13 provides the recall and precision values for different classes (class0 and class1) obtained from the inference of a linear support vector classifier (linear SVC) using different resampling techniques (None, RUS, ROS, SMOTE, SMOTE+TOMEK). The recall measures the ability of a model to correctly identify positive instances, while precision measures the proportion of correctly identified positive instances out of all instances predicted as positive.

The inference results can be summarized as follows:

For class0:

- None: The model achieved a recall of 0.97 and a precision of 0.89.
- RUS (random undersampling): The model achieved a recall of 0.61 and a precision of 0.58.
- ROS (random oversampling): The model achieved a recall of 0.96 and a precision of 1.0.
- SMOTE (synthetic minority oversampling technique): The model achieved a recall of 0.78 and a precision of 0.92.
- SMOTE+TOMEK: The model achieved a recall of 0.79 and a precision of 0.92.

For class1:

- None: The model achieved a recall of 0.13 and a precision of 0.34.
- RUS: The model achieved a recall of 0.62 and a precision of 0.65.
- ROS: The model achieved a recall of 1.0 and a precision of 0.96.
- SMOTE: The model achieved a recall of 0.47 and a precision of 0.22.
- SMOTE+TOMEK: The model achieved a recall of 0.48 and a precision of 0.23.

These values indicate the performance of the linear SVC model using different resampling techniques. It is important to analyze both recall and precision together to assess the effectiveness of the model in correctly identifying positive instances and minimizing false positives.

Regardless to RUS technique results, one can notice that although it is the best method at the time of classification, it reduces the performance, as by decreasing the training data it can discard potentially useful information which could be important for building rule classifiers. Linear SVC is the best performance with RUS. As all of the studied datasets, ROS achieves the best performance. SVC outperforms the other classifiers by 0.99 for all measures. It mitigates the problem of over-fitting caused by random oversampling SMOTE and SMOTE+TOMEK is used to have the effect of the better performance of classifiers fit on the transformed dataset. Both SMOTE+TOMEK and SMOTE seems to be similar with little variations.

6 Discussion

The results of this study suggest that the performance of resampling techniques for cyberbullying datasets depends on several factors, including dataset size, class imbalance ratio, and classifier used. The findings also indicate that no single resampling technique consistently outperforms the others. Therefore, selecting the appropriate resampling technique for a given dataset requires careful consideration of these factors.

One important finding of this study is that classifiers trained on balanced data through resampling are more reliable than those trained on unbalanced data. This underscores the importance of addressing class imbalance in cyberbullying datasets to improve classifier performance. Oversampling and undersampling were found to have different effects on training time, with oversampling leading to longer training times and undersampling reducing them. Therefore, the choice between these two techniques may depend on the available computational resources and time constraints.

Another important finding is that all resampling techniques improve the recall of the minority class and the precision of the majority class when the data are extremely imbalanced. This is particularly important in the context of cyberbullying, where detecting the minority class (i.e., cyberbullying attacks) is crucial for effective prevention and intervention.

Resampling techniques were also found to detect more minority data, especially through oversampling, and improve accuracy by reducing the extent of imbalance. However, it is important to note that resampling cannot improve accuracy if the inaccuracy is not related to imbalance. This highlights the importance of carefully evaluating the reasons for inaccuracies in classifier performance before applying resampling techniques.

The hybrid method SMOTE+TOMEK was found to improve recall and reduce the FN/FP ratio for every classifier, indicating improved sensitivity to cyberbullying. However, the RUS technique, although effective during classification, was found to reduce overall performance by discarding potentially useful information. Linear SVC performed best with RUS, while all studied datasets achieved the best performance with ROS. SVC outperformed other classifiers by 0.99 for all measures. SMOTE and SMOTE+TOMEK were found to mitigate the problem of over-fitting

caused by random oversampling and had similar results with small variations.

7 Concluding remarks

In this paper, we conducted an investigation into the impact of four resampling techniques on the performance of eight classifiers for cyberbullying datasets. Our findings revealed that the effectiveness of resampling techniques is influenced by various factors, including dataset size, imbalance ratio, and the specific classifier employed. No single technique consistently outperformed the others. Notably, classifiers trained on balanced data through resampling exhibited greater reliability compared to those trained on unbalanced data. We observed that oversampling increased training time, while undersampling decreased it. In cases of extreme data imbalance, all resampling techniques enhanced the recall of the minority class and the precision of the majority class. Resampling, particularly through oversampling, facilitated the detection of additional minority data, leading to improved accuracy by reducing the extent of imbalance. However, it is important to note that resampling alone cannot enhance accuracy if the inaccuracies are unrelated to class imbalance. Among the resampling techniques, the hybrid method SMOTE+TOMEK displayed notable improvements in recall and a reduced FN/FP ratio across all classifiers, indicating enhanced sensitivity to cyberbullying instances. Although the RUS technique demonstrated effectiveness in classification, it resulted in an overall performance reduction by discarding potentially valuable information. Linear SVC exhibited the best performance when combined with RUS, while all studied datasets achieved optimal results with ROS. Furthermore, SVC outperformed other classifiers across all measures by a margin of 0.99.

It is essential to acknowledge the limitations of this study, such as the limited exploration of only four resampling techniques and eight classifiers. Other resampling techniques and classifiers may prove effective for cyberbullying datasets. Additionally, our investigation focused solely on binary classification, leaving room for future research to explore multiclass classification for cyberbullying datasets. Furthermore, our study exclusively examined the impact of resampling techniques on cyberbullying datasets, while future work should investigate their effectiveness on other types of datasets with class imbalance. To expand on the research in this field, we propose several avenues for future work. Firstly, the exploration of alternative resampling techniques and classifiers specifically tailored to cyberbullying datasets would be valuable. Secondly, investigating multiclass classification methods could enhance classifier performance for cyberbullying datasets. Thirdly, extending the investigation to other types of datasets with class imbalance, such as medical datasets, would provide broader insights. Additionally, combining multiple resampling techniques could be explored to further improve classifier performance. Lastly, the effectiveness of combining resampling techniques with other approaches, such as feature selection or extraction, for enhancing classifier performance in cyberbullying datasets warrants investigation.

Authors contributions This work was carried out in collaboration among all authors. All authors designed the study, performed the statistical analysis, and wrote the protocol. Authors MK, TMM, and TAEH managed the analyses of the study, managed the literature searches, and wrote the first draft of the manuscript. All authors read and approved the final manuscript.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB). Not applicable.

Data availability Https://www.kaggle.com/datasets/saurabhshahane/cyberbullying-dataset.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical statement This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Abdellatif S, Ben Hassine MA, Ben Yahia S, and Bouzeghoub A, ARCID: a new approach to deal with imbalanced datasets classification," in SOFSEM 2018: Theory and Practice of Computer Science: 44th International Conference on Current Trends in Theory and Practice of Computer Science, Krems, Austria, January 29-February 2, 2018, Proceedings 44, Springer, 2018, pp. 569–580.
2. Ali A, Shamsuddin SM, and Ralescu AL (2015), Classification with class imbalance problem: a review," Int J Adv. Soft Compu Appl, 7(3).

3. Khairy M, Mahmoud TM, Abd El-Hafeez T (2021) Automatic detection of cyberbullying and abusive language in Arabic content on social networks: a survey. Procedia Comput. Sci. 189:156–166
4. Colton D, Hofmann M (2019) Sampling techniques to overcome class imbalance in a cyberbullying context. J Comput-Assist Linguist Res 3(3):21–40
5. Omar A, Mahmoud TM, Abd-El-Hafeez T, Mahfouz A (2021) Multi-label arabic text classification in online social networks. Inf Syst 100:101785
6. Ali B, O'Sullivan D (2020) Cyberbullying severity detection: a machine learning approach. PLoS ONE 15:e0240924. https://doi.org/10.1371/journal.pone.0240924
7. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
8. Yen S-J, Lee Y-S (2009) Cluster-based under-sampling approaches for imbalanced data distributions. Expert Syst Appl 36 (3):5718–5727
9. Soda P (2011) A multi-objective optimisation approach for class imbalance learning. Pattern Recognit 44(8):1801–1810
10. Liu AY (2004), The effect of oversampling and undersampling on classifying imbalanced text datasets.
11. Naseriparsa M, Bidgoli A, and Varaee T (2014), "Improving Performance of a Group of Classification Algorithms Using Resampling and Feature Selection," *ArXiv Prepr. ArXiv14031946*.
12. Khaldy MA, Kambhampati C (2018) Resampling imbalanced class and the effectiveness of feature selection methods for heart failure dataset. Int Robot Autom J 4(1):1–10
13. Hacibeyoglu M and Ibrahim MH (2018), The effect of over-sampling and under-sampling techniques in medical datasets, in International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES'18), 2018.
14. Talpur BA and O'Sullivan D (2020), Multi-class imbalance in text classification: A feature engineering approach to detect cyberbullying in twitter, in Informatics, MDPI, 2020, p. 52.
15. Chkifa A and Dolbeault M (2023), Randomized least-squares with minimal oversampling and interpolation in general spaces, *ArXiv Prepr. ArXiv230607435*.
16. Liu SM, Chen J-H, Liu Z (2023) An empirical study of dynamic selection and random under-sampling for the class imbalance problem. Expert Syst Appl 221:119703
17. Elreedy D, Atiya AF and Kamalov F (2023), A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning, Mach. Learn., pp. 1–21, 2023.
18. Dey I, and Pratap V (2023), A comparative study of SMOTE, borderline-SMOTE, and ADASYN oversampling techniques using different classifiers," in 2023 3rd International Conference on Smart Data Intelligence (ICSMDI), IEEE, 2023, pp. 294–302.
19. Chandra W, Suprihatin B, Resti Y (2023) Median-KNN Regressor-SMOTE-Tomek links for handling missing and imbalanced data in air quality prediction. Symmetry 15(4):887
20. Fu S, Tian Y, Tang J, Liu X (2023) Cost-sensitive learning with modified Stein loss function. Neurocomputing 525:57–75
21. Reynolds K, Kontostathis A, and Edwards L (2011), Using machine learning to detect cyberbullying," in 2011 10th International Conference on Machine learning and applications and workshops, IEEE, 2011, pp. 241–244.
22. Dinakar K, Jones B, Havasi C, Lieberman H, Picard R (2012) Common sense reasoning for detection, prevention, and mitigation of cyberbullying. ACM Trans Interact Intell Syst TiiS 2(3):1–30
23. Nahar V, Li X, Pang C and Zhang Y (2013), Cyberbullying detection based on text-stream classification, in The 11th Australasian Data Mining Conference (AusDM 2013), 2013.
24. Dadvar M, Trieschnigg D, Ordelman R, and De Jong F (2013), Improving cyberbullying detection with user context, in Advances in Information Retrieval: 35th European Conference on IR Research, ECIR 2013, Moscow, Russia, March 24–27, 2013. Proceedings 35, Springer, 2013, pp. 693–696.
25. Feng W, Huang W, Ren J (2018) Class imbalance ensemble learning based on the margin theory. Appl Sci 8(5):815
26. Chavan VS and Shylaja SS, Machine learning approach for detection of cyber-aggressive comments by peers on social media network," in *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, 2015, pp. 2354–2358.
27. Mangaonkar A, Hayrapetian A, and Raje R, Collaborative detection of cyberbullying behavior in Twitter data, in 2015 IEEE International Conference on Electro/Information Technology (EIT), IEEE, 2015, pp. 611–616.
28. Van Hee C et al. (2015), Detection and fine-grained classification of cyberbullying events, in Proceedings of the International Conference Recent Advances in Natural Language Processing, 2015, pp. 672–680.
29. Ptaszynski M et al (2016) Sustainable cyberbullying detection with category-maximized relevance of harmful phrases and double-filtered automatic optimization. Int J Child-Comput Interact 8:15–30
30. Singh VK, Huang Q, and Atrey PK (2016), Cyberbullying detection using probabilistic socio-textual information fusion, in 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2016, pp. 884–887.
31. Al-Garadi MA, Varathan KD, Ravana SD (2016) Cybercrime detection in online communications: the experimental case of cyberbullying detection in the Twitter network. Comput Hum Behav 63:433–443
32. Zhao R, Zhou A, and Mao K (2016), Automatic detection of cyberbullying on social networks based on bullying features," in Proceedings of the 17th International Conference on Distributed Computing and Networking, 2016, pp. 1–6.
33. Sugandhi R, Pande A, Agrawal A, Bhagat H (2016) Automatic monitoring and prevention of cyberbullying. Int J Comput Appl 8:17–19
34. Hosseinmardi, H, Rafiq RI, Han R, Lv Q, and Mishra S, Prediction of cyberbullying incidents in a media-based social network, in 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2016, pp. 186–192.
35. Zhang X et al. (2016), "Cyberbullying detection with a pronunciation based convolutional neural network," in 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, 2016, pp. 740–745.
36. Rosa H et al (2019) Automatic cyberbullying detection: a systematic review. Comput Hum Behav 93:333–345
37. Haidar B, Chamoun M, Serhrouchni A (2017) A multilingual system for cyberbullying detection: Arabic content detection using machine learning. Adv Sci Technol Eng Syst J 2(6):275–284
38. Haidar B, Chamoun M, and Serhrouchni A (2018) Arabic cyberbullying detection: Using deep learning," in *2018 7th International Conference on Computer and Communication Engineering (iccce)*, IEEE, 2018, pp. 284–289.
39. Haidar B, Chamoun M, and Serhrouchni A (2019), Arabic cyberbullying detection: enhancing performance by using ensemble machine learning," in 2019 international conference on internet of things (ithings) and ieee green computing and communications (greencom) and ieee cyber, physical and social computing (cpscom) and ieee smart data (smartdata), IEEE, 2019, pp. 323–327.
40. Mouheb D, Abushamleh MH, Abushamleh MH, Al Aghbari Z, and Kamel I, Real-time detection of cyberbullying in arabic twitter streams, in 2019 10th IFIP International Conference on New

Technologies, Mobility and Security (NTMS), IEEE, 2019, pp. 1–5.
41. Mouheb D, Albarghash R, Mowakeh MF, Al Aghbari Z, and Kamel I, Detection of Arabic cyberbullying on social networks using machine learning, in 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), IEEE, 2019, pp. 1–5.
42. AlHarbi BY, AlHarbi MS, AlZahrani NJ, Alsheail M, Alshobaili J, Ibrahim DM (2019) Automatic cyber bullying detection in Arabic social media. Int J Eng Res Technol 12(12):2330–2335
43. Rachid BA, Azza H, and Ghezala HHB (2020) Classification of cyberbullying text in Arabic, in 2020 International Joint Conference on Neural Networks (IJCNN), IEEE, 2020, pp. 1–7.
44. Kanan T, Aldaaja A, Hawashin B (2020) Cyber-bullying and cyber-harassment detection using supervised machine learning techniques in Arabic social media contents. J Internet Technol 21 (5):1409–1421
45. Farid D, El-Tazi N (2020) Detection of cyberbullying in tweets in Egyptian dialects. Int J Comput Sci Inf Secur IJCSIS 18(7):34–41
46. AlHarbi BY, AlHarbi MS, AlZahrani NJ, Alsheail MM, Ibrahim DM (2020) Using machine learning algorithms for automatic cyber bullying detection in Arabic social media. J Inf Technol Manag 12(2):123–130
47. Hilario, AF, López SG, Galar M, Prati RC, Krawczyk B, and Herrera F (2018) Learning from imbalanced data sets, *Artif. Intell. Springer Cham*, 2018.
48. M. Khairy, T. M. Mahmoud, and T. Abd El-Hafeez, "The Effect of Rebalancing Techniques on the Classification Performance in Cyberbullying Datasets," 2022.
49. "Cyberbullying Dataset | Kaggle." https://www.kaggle.com/datasets/saurabhshahane/cyberbullying-dataset (accessed Jul. 25, 2023).
50. Van Hulse J, Khoshgoftaar TM, and Napolitano A (2007) "Experimental perspectives on learning from imbalanced data," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 935–942.
51. Kubus M (2020) Evaluation of resampling methods in the class unbalance problem. Econometrics 24(1):39–50
52. Learning I (2013) Foundations, algorithms, and applications. Wiley 10:9781118646106
53. Brownlee J (2020), Imbalanced classification with Python: better metrics, balance skewed classes, cost-sensitive learning. Machine Learning Mastery, 2020.
54. Boardman J, Biron K, and Rimbey R (2018), Mitigating the effects of class imbalance using SMOTE and Tomek Link Undersampling in SAS®," in SAS Global Forum.
55. Ajiboye AR, Abdullah-Arshah R, and Hongwu Q (2015) "Evaluating the effect of dataset size on predictive model using supervised learning technique,".
56. Sordo M and Zeng Q (2005), On sample size and classification accuracy: a performance comparison, in International Symposium on Biological and Medical Data Analysis, Springer, pp. 193–201.
57. García V, Sánchez JS, and Mollineda RA, Exploring the performance of resampling strategies for the class imbalance problem, in Trends in Applied Intelligent Systems: 23rd International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2010, Cordoba, Spain, June 1–4, 2010, Proceedings, Part I 23, Springer, 2010, pp. 541–549.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.