Received: 23 October 2023 | Revised: 27 November 2023 | Accepted: 13 December 2023 | Published online: 18 December 2023

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2024, Vol. 3(3) 226–239
DOI: 10.47852/bonviewJCCE32021924

BON VIEW PUBLISHING

An Effective Approach for Multiclass Classification of Adverse Events Using Machine Learning

Indu Bala[1,*], Thu-Lan Kelly[2], Renly Lim[2], Marianne H. Gillam[2] and Lewis Mitchell[1]

[1]*School of Computer and Mathematical Sciences, The University of Adelaide, Australia*

[2]*Clinical and Health Sciences, University of South Australia, Australia*

Abstract: Implantable medical devices are commonly used to treat various medical conditions. These devices, however, may cause serious adverse events, including repeated surgical intervention and death. Prolonged use of some implantable medical devices can shorten life expectancy and significantly decrease a person's quality of life. Large adverse event databases can be used to predict serious adverse events by training machine learning (ML) models on available data. However, the large volume of data and long free-text response make it challenging to use the databases effectively. This study focuses on one such dataset: the Australian Database of Adverse Event Notifications, comprising text written by patients, or healthcare professionals, or pharmaceutical industry. The study focuses on predicting three significant events: Injury, No Injury, and Death, based on the adverse events reported about the implanted device. A new ML approach called the random regression voting classifier, which combines random forest (RF) and logistic regression (LR), is proposed. The model's efficiency is evaluated through experiments using techniques, such as Bag of Words, Term-Frequency-Inverse-Document Frequency, and Global Vector, and is compared to existing ML models such as decision tree, RF, kernel support vector machine, Naive Bayes, LR, and XGboost. The results demonstrate a higher performance in predicting adverse events than other considered approaches. The various experimental analyses showed that the proposed approach performed better than other ML models.

Keywords: post-market surveillance, adverse event, implantable medical device, machine learning, natural language processing

1. Introduction

The World Health Organization (WHO) defines a medical device as "any instrument, apparatus, implement, machine, appliance, implant, reagent for in vitro use, software, material, or another similar or related article, intended by the manufacturer to be used, alone or in combination, for human beings, for one or more specific medical purpose" [1]. These devices play a crucial role in preventing, diagnosing, treating, and rehabilitating illnesses and diseases in a safe and effective manner. Examples of medical devices include surgical equipment, pacemakers, mesh, incubators, and implants. According to the WHO, there are approximately 2 million different types of medical devices available on the global market. Prior to their release onto the market, these devices undergo thorough testing. However, post-market surveillance reveals a significant number of unfavorable symptoms caused by implant devices every year [1]. Adverse medical device events (AMDEs), specifically those associated with higher-risk medical devices that significantly impact patient outcomes, have captured widespread attention and prompted calls for action to establish approaches that effectively address both access to innovative medical devices and the need to minimize risks, ensuring enhanced patient safety [2]. Post-market surveillance is a valuable approach that entails the systematic collection and analysis of a substantial amount of data over a period to detect any potential adverse events associated with medical devices [3]. In this regard, the data collected through spontaneous reporting play a critical role in evaluating product performance and ensuring patient safety. Spontaneous reporting relies on voluntary reports of adverse events or side effects from healthcare professionals, patients, or consumers and is an important tool for identifying previously unknown or unanticipated adverse events associated with medical devices. However, the effectiveness of this approach relies heavily on the quality of the information provided by healthcare professionals and consumers in their reports.

A complete adverse event report may include several key pieces of information, such as the product name, mode and serial numbers (in the case of medical devices), demographic data, and a concise clinical description of the adverse event [1, 4]. Additionally, temporal information, including the date of event onset and start/stop dates for the use of the medical product, the dose/frequency of use (if applicable), and outcome, can also be included. However, the dataset generated by post-market surveillance is often enormous, and the information provided may be in an unstructured format, which can make extracting data on adverse events more complicated. However, if the ability to interpret, analyze, and utilize unstructured medical data is enhanced, increased benefits can be achieved in terms of patient

*Corresponding author: Indu Bala, School of Computer and Mathematical Sciences, The University of Adelaide, Australia. Email: indu.bala@adelaide.edu.au

treatment, public health management, and medical research [5]. This task can be accomplished manually but will require significant resources. Therefore, a robust methodology for interpreting and analyzing the data is required. The most common and effective way to deal with unstructured data is through the use of machine learning (ML) techniques.

ML is a self-regulated discovery of potentially valid or useful knowledge and novel hidden patterns from datasets [6]. ML techniques have been widely applied in the healthcare sector for forecasting patient outcomes and evaluating medical devices [7]. The clinical reports generally contain a high level of noise, sparsity, complex medical vocabularies, medical measures, abbreviations, misspelled words, and poor grammatical sentences [8, 9]. In the context of data analysis, noisy data refer to data that contain a substantial amount of irrelevant or meaningless information, which is commonly referred to as noise. This can manifest as data corruption or other forms of errors that render the data inaccurate or incomplete. So, the first task for text classification is reducing the noise from the dataset. To address this issue, Nguyen and Patrick [9] developed a lexical technique and a preanalysis of corpora to reveal the linguistic characteristics. Subsequently, many predictive modeling approaches have been developed to analyze large datasets and better predictions. Likewise, Zhang et al. [10] presented a method for semantic role labeling (SRL) of clinical text. SRL aims to identify the different roles that different parts of a sentence play in the overall meaning. The authors evaluated their approach using three state-of-the-art parsers and compared their performance on a clinical text corpus. The work provided a formal evaluation of SRL performance on clinical text, which has unique challenges compared to other types of text due to the specialized language used in healthcare. Similarly, Renganathan [11] presented text mining approaches with a focus on biomedical clustering. The author presented a method for partitioning text into small clusters before testing.

Later, Turner et al. [12] automated manual chart review of doctors' notes using text classifiers based on NLP techniques and pattern recognition ML algorithms. The authors evaluated the performance of traditional classifiers for identifying patients with systemic lupus erythematosus (SLE) compared to a newer Bayesian word vector method. Moreover, Tripoliti et al. [13] used ML techniques for predicting heart diseases, and Forsyth et al. [14] used ML algorithms to extract patient-reported symptoms from unstructured notes in electronic health records. It showed how NLP techniques can be applied to extract meaningful information from free-text notes, which can be difficult to process using traditional methods. In addition, Yang et al. [15] used a transfer learning approach to classify electronic discharge summaries based on their clinical concepts. The method achieved high accuracy and reduced the need for manual feature engineering. Alsentzer et al. [16] explored BERT models for clinical text and demonstrated that using domain-specific models yields performance improvements on three common clinical NLP tasks compared to nonspecific embeddings. This work highlighted the importance of using domain-specific models for clinical text, which has unique challenges compared to other types of text. Later, Martenot et al. [17] developed an augmented intelligence methodology for automatically identifying relevant publications mentioning an established link between a drug and a serious adverse event. This work demonstrated the potential for NLP techniques to be used in drug safety monitoring, which is critical for ensuring patient safety.

In short, the need for implant devices to enhance health can potentially lead to more significant issues if not in advance assessed. Proactive decision-making is invaluable, and delving into the analysis of unstructured free-text inputs from patients during treatment emerges as a promising avenue for making informed decisions. This serves as the primary motivation for our study. Our objective is to develop a robust ML technique that predicts adverse events by scrutinizing patients' reports, anticipating outcomes that could pose serious risks. This predictive insight holds the potential to empower healthcare organizations in making informed decisions regarding medical devices, anticipating, and preventing serious outcomes.

This study makes the following contributions:

1) An overview of efficient methods to investigate the adverse events due to implanted medical devices leading to serious injury or death. The prognosis is based on three significant events including "No Injury", "Injury", and "Death".
2) A vote-based classifier random regression voting classifier (RRVC) is devised that combines the approach of random forest (RF) and logistic regression (LR) classifier under soft voting criterion.
3) To analyze the influence of data balancing on the Australian Database of Adverse Event Notifications (DAEN) dataset, the performance of unbalanced dataset results is compared with the balanced dataset.

The paper is organized as follows: In Section 2, "Materials and Methods" presents the DAEN dataset description, ML-based models, and the proposed approach. In Section 3, "Result Analysis and Discussion" describes the experiments and outcomes. Finally, the study is concluded in Section 4.

2. Materials and Methods

The aim of this study is to predict the risk of serious injury or death in patients who have reported adverse events following the implantation of a medical device. The study uses a multiclass classification method to categorize the reported adverse events in the dataset as "Injury", "No Injury", or "Death". The following section provides a brief overview of the dataset and the methodology used in this study.

2.1. Dataset description

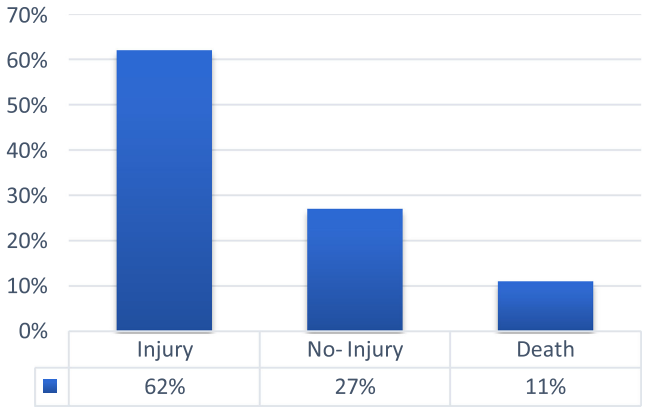
The Australian DAEN dataset contains spontaneous reports of medicines and medical devices and is maintained by the Therapeutic Goods Association (TGA). The TGA is the regulatory body responsible for evaluating, assessing, regulating, and monitoring medicines, medical devices, and biological to ensure public health and safety. This study utilizes the DAEN database, which has not been extensively studied for medical devices. The data are available on the Therapeutic Goods Administration website [18]. The data, which were extracted from reported adverse events regarding implanted medical devices between 2013 and 2019, consist of 21,381 elements (after removing duplicate rows) and includes 17 columns, as detailed in Table 1. This study focuses on investigating the risk of death in patients due to implanted medical devices and uses the "Reported source category" and "Event description" variables for this purpose. The "Reported source category" column includes "Injury", "No-Injury", "Death", and "Not Known" cases, while the "Event description" column includes patients' reviews during treatment and symptoms. However, the "Not Known" category in the "Reported source category" variable

Table 1
Data description

Variables	Descriptions
Report number	Identification of each patient case
Report date	Receiving date of adverse events report
Trade name	Name of the trader of medical device
Sponsors	Sponsor company of the device
Manufacturer	Manufacturer of the device
ARTG number	The register number of Australian register of Therapeutic Goods (ARTG)
GMDN term	International nomenclature for describing medical devices.
Device classification	Class of device
Sterile	Use of sterile during treatment (Y/N)
Single use	Use of sterile once or more (Y/N)
Model number	Device model number
Software version	Software version details
Reported event outcome	Outcome of adverse event during treatment
Reported source category.	Adverse event reported source.
Event type	Type of medical device
Event description	Reported symptoms and reviews.
Other medical devices reported as being used	Another device is being used during treatment (Y/N)

is not informative and represents only 2% of the data. Therefore, it is excluded from the analysis, and the model is trained on the remaining three meaningful categories. The distribution of these categories is presented in Figure 1.

Figure 1
Class distribution



2.2. Problem statement

In this study, our focus is the analysis of reported adverse events post-medical device implantation. Recognizing that certain side effects may be transient while others pose the risk of severe problems, our study aims to pinpoint potential adverse events, specifically serious injuries or deaths resulting from medical device implantation. To accomplish this objective, we employ a multiclass classification approach, categorizing events into Death, Injury, or No Injury. This study provides an improved model to predict the likelihood of adverse events based on patient data, with the aim of aiding the TGA to identify potential issues early and take appropriate action.

2.3. Methodology

Initially, the DAEN dataset was obtained from the TGA website, comprising of 92702 reports. However, the data were found to be highly inconsistent and contained a significant number of duplicate entries. Therefore, a data preprocessing technique was implemented to improve the accuracy of the classification results. The experiments were conducted using the Python platform.

2.3.1. Data preprocessing

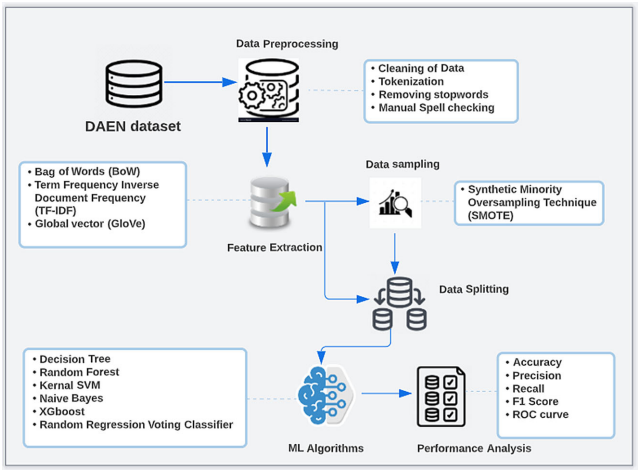
1) **Spell checking**: To ensure the accuracy of the data used in this study, an automated spell-checking software was utilized to identify and correct any misspelled words within the medical database consisting of patient reviews. The spell-checker lexicon was consulted for each word in the text dataset [19, 20]. If a word was not found in the dictionary, an error was detected. However, as the data may contain errors that the automated software is unable to fix, manual spell checking was also employed. This approach presented a set of possible options for any misspelled words and enabled the user to make the most appropriate selection within the context of the reports.
2) **Tokenization**: In this step, the process of breaking down a long text into smaller units, known as tokens (words), is applied to the "Event Description" containing the patient's review.
3) **Clean data**: In this step, the data undergo a thorough cleaning process to prepare it for natural language processing (NLP) tasks. Two methods are employed: converting all text data to lowercase and removing all special characters and symbols. This ensures that the data are in a standard and easily comprehensible format for the machine. Additionally, any missing values are handled appropriately, and duplicate and empty rows are eliminated, resulting in a final dataset of 21381 entries.
4) **Removing stop words and irrelevant English words**: In this step, we eliminated those words that had minimal contributions, such as common pronouns and articles, as well as medical measurements; they are {"patient", "mg", "doctors", "B", "case", "always", "getting"} omitted to streamline the data and improve the accuracy of our analysis.

2.3.2. Feature extraction

In this step, the text data have been cleaned and prepared for ML techniques. To achieve this, a technique called feature extraction is utilized to extract the most relevant and significant features from the preprocessed data, to improve the performance of the model [21]. Three different feature extraction techniques have been used in this study, including Bag of Words (BoW), Global Vector for Word Representation (GloVe), and Term-Frequency-Inverse Document Frequency (TF-IDF). BoW is a method of converting text data into numerical features, where each word is considered as a feature, regardless of the number of times it appears in the documents [22]. TF-IDF is a technique that calculates the weight of each word, which shows the importance of that word in the text. The weight is determined by two metrics: Term frequency (TF) and inverse document frequency (IDF) [23]. GloVe, on the other hand, generates word embedding of the document by mapping the relationship between words, which is achieved by aggregating global co-occurrence matrices [24]. This technique groups similar words together and rejects different words based

on the co-occurrence matrix of the corpus. The framework of devised process is presented in Figure 2.

Figure 2
Architecture of the methodology devised



2.3.3. Machine learning models

We utilized ML algorithms commonly discussed in the literature. Detailed descriptions of the employed algorithms and their hyperparameters used in this study are discussed as follows:

1) **Decision Tree (DT)**: A DT is a popular supervised ML algorithm used for both classification and regression tasks. It works by recursively partitioning the training data into smaller subsets based on the values of the input features, until a stopping criterion is met. The tree structure consists of internal nodes that represent a feature or attribute, and branches that represent the possible values of the feature. The leaves of the tree correspond to the predicted class label or output value. During training, the algorithm determines the optimal split at each node by maximizing the information gain or minimizing the impurity of the subsets [25]. One of the advantages of DTs is their interpretability, as the resulting tree can be easily visualized and understood by humans. DTs are also robust to noisy data and can handle both categorical and numerical features. They can also handle missing data by using surrogate splits. This study involves tuning various hyperparameters of the algorithm to improve its predictive power and performance, as well as to speed up its execution. The values of the max_depth, and random_state hyper parameters are adjusted according to the specific requirements. The max_depth parameter is used to improve the predictive power of the algorithm by limiting the maximum depth of tree. It limits the number of nodes in the tree and can help prevent overfitting. We set it to 300 for this study. The random_state parameter is set to 50 to control the randomness of the samples and ensure that the model produces consistent outcomes, which also helps to enhance the computational speed of the algorithm.
2) **Random Forest (RF)**: The RF classification algorithm is a versatile method that can be used for both regression and classification tasks [26]. It is a supervised learning algorithm that builds a multitude of DTs and combines their outputs to produce a final prediction. The main idea behind RF is to create a set of DTs that are independent of each other, but when combined, provide a more accurate and robust prediction. Each DT in the forest is constructed by selecting a random subset of the training data and a random subset of features. This process helps to reduce overfitting and improve the generalization of the model [27]. One of the advantages of RF is that it can handle high-dimensional data and large datasets with many features. It is also a nonparametric model, which means it can capture complex nonlinear relationships between features and the target variable. The n_estimator parameter represents the number of DTs to be assembled by the algorithm before making predictions. For this study, a value of 100 is set for this parameter to obtain highly accurate results, as this value is considered to be the number of weak learners in the algorithm. The max_depth and random_state set to 300 and 50, respectively.
3) **Multinomial Naive Bayes Model (MNB)**: The Multinomial Naive Bayes (MNB) model is a probabilistic model used for text classification tasks. It is based on the Bayes theorem, which states that the probability of a hypothesis given evidence is proportional to the probability of the evidence given the hypothesis, multiplied by the prior probability of the hypothesis. In the case of text classification, the hypothesis is the class label of a document, and the evidence is the set of words or features that occur in the document. The MNB model assumes that the frequency of occurrence of each word or feature in a document follows a multinomial distribution. It also assumes that the occurrence of each word or feature is independent of the occurrence of all other words or features in the document, which is known as the "naive" assumption. To train the MNB model, the frequency of occurrence of each word or feature is computed for each class in the training data. Then, the probabilities of each word or feature occurring in each class are calculated using the maximum likelihood estimation (MLE) method. These probabilities are used to compute the conditional probability of a document belonging to each class, given the observed set of words or features in the document [28]. To classify a new document, the MNB model computes the conditional probability of the document belonging to each class, given the observed set of words or features in the document, using the probabilities computed during training. The class with the highest probability is then assigned as the predicted class for the document. The hyperparameters set as random_state = 50, multi_class = "ovr".
4) **Logistic Regression (LR)**: The LR algorithm is a statistical ML classifier that maps input features to discrete target variables by estimating probabilities using a sigmoid function. This function constrains the probability values between the target variables, forming an "S"-shaped curve. LR is widely used for efficient classification tasks and represents an advanced version of linear regression, applicable to complex linear and nonlinear datasets for both classification and prediction tasks. One of its common applications is modeling binary data. The LR technique involves multiplying input values with weight values. It is renowned for its effectiveness in detecting defaulters and is a popular choice among ML classifiers due to its simplicity and reliance on fewer assumptions [29, 30]. In this study, the LR algorithm utilized the "saga" solver for faster computation and improved results when handling large datasets. The "multi_class" parameter was set to "ovr" as it excels in binary classification tasks. The inverse regularization parameter "C" was assigned a value of 3.0, which is inversely proportional to the Lambda regulator and determines the strength of the regularization. This parameter aids in reducing the risk of model overfitting, with smaller values indicating stronger regularization.
5) **Support Vector Machine (SVM)**: It belongs to parametric test family and can be applied for regression and classification.

However, it is considered best for classification problem [15]. In SVM algorithm, a line is drawn in form of hyperplane that divides a feature space into two subspaces. One subspace contains vectors that belong to a category, and another contains vector that do not belong to that category. The optimal hyperplane is the one with the largest distance between two subspaces called as maximum margin hyperplane [29]. If the data are nonlinear and complex, then kernel function can be used to form the multidimensional hyperplane. The choice of kernel depends on the complexity and features of the dataset, which makes understanding the data crucial for achieving good accuracy and performance. In this study, the radial basis function kernel was used along with C = 1.0, epsilon = 0.2.

2.3.4. Data sampling using Synthetic Minority Oversampling Technique (SMOTE)

SMOTE is a popular data augmentation technique used in ML and data science to balance class distribution in imbalanced datasets, such as when the minority class is significantly smaller than the majority class. The basic idea behind SMOTE is to create synthetic samples of the minority class by interpolating between existing minority class instances. The algorithm works by selecting an instance from the minority class and finding its k nearest neighbors in the feature space. It then creates new instances by interpolating between the selected instance and its k nearest neighbors, effectively creating new instances in the feature space [31]. The SMOTE algorithm has several parameters that can be tuned to control the number and distribution of synthetic samples generated. The most important parameters are the number of nearest neighbors' k to consider, and the amount of interpolation to use when generating new samples. Other parameters include the random seed used for reproducibility, and the balance ratio, which specifies the desired ratio of minority to majority class instances after augmentation. We set hyper parameters as: sampling_strategy = "str", random_state = 50, weight = "equal", k = 5, n_features = 20, n_clusters_per_class = 1.

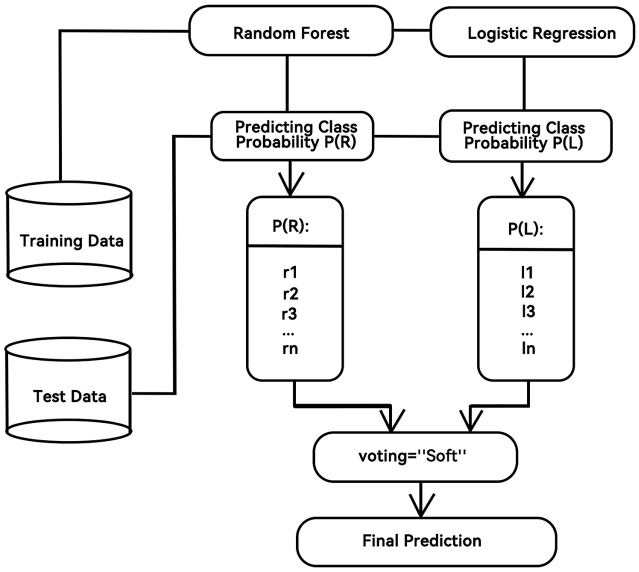
2.4. Proposed random regression voting classifier (RRVC)

We proposed a new ensemble method RRVC that combines the predictions of two individual classifiers, RF and LR, to generate a final output. The reason for suggesting RRVC is to take advantage of the ensemble capabilities of RF and LR and to adjust the parameters to decrease the error.

The leverage of RF used in this proposed RRVC lies in its learning strategies, which involve creating multiple DTs and combining their predictions to make a final decision. RF is particularly useful for large datasets as it can determine the significance of each feature and select the most important ones. It also deals effectively with high-dimensional data, making it suitable for multiclass classification. The randomizing property of RF helps prevent overfitting, ensuring that the model generalizes well to unseen data. On the other hand, we leverage LR for its successful implementation, extensively used for binary and multiclass classification in comparison to statistical methods. LR provides probability predictions and is computationally less expensive. It also offers built-in flexibility, enabling fine-tuning of the model's fit or reduction of errors through regularization parameters. The detailed implementation is illustrated in Figure 3.

The proposed RRVC classifier also uses a soft voting scheme to generate a final prediction [4, 20]. Soft voting classifies input data based on the probabilities of all the predictions made by classifiers.

Figure 3
Framework of random regression voting classifier



This means that the final prediction is obtained by averaging the probability p given by both the RF and LR classifiers to each target class. This approach helps to select incidents from the data that contain valuable information, enhancing accuracy. The RRVC framework is described in Algorithm 1. It also incorporates similar parameter tuning described in the Section 2.3.3.

We can compute the target class for the weights assigned to predictions $r_1, r_2, r_3, \ldots, r_n$ made by classifier RF and $l_1, l_2, l_3, \ldots, l_n$ by classifier LR, respectively, as

$$P(R) = r_1, r_2, r_3, \ldots, r_n \tag{1}$$

$$P(l) = l_1, l_2, l_3, \ldots, l_n \tag{2}$$

$$Final\ Prediction = argmax \sum_{i=0}^{n} P(R) + P(L) \tag{3}$$

where $P(R)$ *and* $P(L)$ are predictions made by RF and LR, respectively. Soft voting classifies input data based on the probabilities of all the predictions made by classifiers. Weights applied to each classifier in the soft voting scheme get applied appropriately based on the Equation (3). The weight parameter controls the contribution of each classifier to the final prediction therefore we set equal weights criteria for this study so that all classifiers are given equal importance in making the final prediction. Let us understand this using an example. For a particular record, the two classifiers RF and LR make the following predictions in terms of probabilities in favor of classes [Injury, No Injury, Death]:

$$RF- > [0.1, 0.3, 0.6], LR- > [0.3, 0.3, 0.4]$$

With equal weights, the probabilities will get calculated as the following:

$$\text{Prob of Class 'Injury'} = 0.5 * 0.1 + 0.5 * 0.3 = 0.2$$

$$\text{Prob of Class "No Injury"} = 0.5 * 0.3 + 0.5 * 0.3 = 0.3$$

$$\text{Prob of Class "Death"} = 0.5 * 0.6 + 0.5 * 0.4 = 0.5$$

The probability predicted by RRVC will be [20, 30, and 50%]. The class will most likely be "Death".

In summary, RRVC is an influential classifier that combines the strengths of both RF and logistic regression. It utilizes the feature selection capability of RF and the probability prediction and regularization flexibility of LR to generate accurate and robust predictions.

Algorithm 1: Proposed approach: Random regression voting classifier

Random regression voting classifier (RRVC)

Input: *Event description*
Output: *Patient having implanted device ← Serious injury or no injury or death*
Procedure: Data splitting
Training_set = (Event description, labels)
Testing_set = (Event description)
Training_set, testing_set
Procedure: Voting classifier
Voting_criterion = "soft (set equal weights criteria and threshold = 0.5)
R = Random_forest (Training_set)
L =Logistic regression (Training_set)
Procedure: Predictions made by R
$P(R) \leftarrow Testing\_set$
$P(R) = r_1, r_2, r_3, \ldots, r_n$
Procedure: Predictions made by S
$P(L) \leftarrow Testing\_set$
$P(L) = l_1, l, l_3, \ldots, l_n$
$$Final\ Prediction = argmax \sum_{i=0}^{n} P(R) + P(L)$$

3. Results Analyses and Discussion

In this section, we conduct experiments and analyze the results of various methods, comparing their performance with the proposed RRVC using various performance measures.

3.1. Evaluation measures

In this study, we employed cross-validation techniques to evaluate the performance of the considered algorithms, ensuring the robust generalization of the model, mitigating bias, and preventing overfitting. A fivefold cross-validation technique was adopted for this purpose, involving iterative training and evaluation with different combinations of training and test sets. This approach allows for a comprehensive assessment of the model's performance, enhancing reliability by averaging results across multiple iterations [4]. The dataset was divided into two parts using an 80:20 ratio [32].

Moreover, the proposed algorithm's effectiveness was evaluated using multiple metrics, including accuracy, F-Measure, precision, and recall [33]. Additionally, we employed the receiver operating characteristic (ROC) curve and the area under the ROC curve (AUC) to evaluate the algorithm's performance. The ROC curve displays the classification model's performance at all classification thresholds by plotting the false-positive rate (FPR) against the true-positive rate (TPR) [34]. AUC provides an aggregate measure of all possible classification thresholds and can be interpreted as the probability that the model ranks a random positive example higher than a random negative example. In summary, we used a combination of these diverse evaluation metrics to comprehensively evaluate the effectiveness of the proposed RRVC approach.

3.2. Result analysis

The six most used algorithms in the literature were compared to the proposed RRVC approach in terms of multiclass classification in two distinct scenarios. In the first scenario, the data had an imbalanced class distribution, while in the second scenario, the data were made balanced by using oversampling techniques SMOTE. The results of these two scenarios will be discussed separately.

3.2.1. Results analysis for scenario 1

In this case, the data used in the experiment are imbalanced, with more instances of injury classes than others. The proposed classifier, RRVC, was compared with other ML algorithms to determine its effectiveness. The results of this comparison were presented in tables, with Table 2 showing the results when using TF-IDF as the feature extraction method. In this table, the proposed classifier attained the highest accuracy of 84%. Other algorithms such as XGboost and RF also performed well, with accuracy rates of 80%.

Table 3 shows the results when using BoW as the feature extraction method. In this table, the proposed RRVC scored a maximum accuracy of 79%. However, the use of BoW did not improve the results of the classifiers, and, in fact, a significant drop in performance was observed for most of the ML algorithms. The exception to this was the DT classifier, which had a 2% increase in performance compared to the TF-IDF results.

Table 2
Classification results of ML algorithms using TF-IDF without SMOTE (Scenario-1)

ML-Algorithms	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.77	0.44	0.56	80
	Injury	0.83	0.86	0.81	
	No-injury	0.81	0.64	**0.67**	
Decision tree	Death	0.44	0.42	0.43	69
	Injury	0.84	0.83	0.84	
	No-injury	0.58	0.59	0.58	
Naïve Bayes	Death	0.53	0.62	0.56	70
	Injury	0.84	0.52	0.66	
	No-injury	0.47	0.68	0.56	
Kernel SVM	Death	0.83	0.12	0.43	79
	Injury	0.84	0.94	0.89	
	No-injury	0.74	0.53	0.62	
Logistic regression	Death	0.13	0.59	0.51	72
	Injury	**0.93**	0.69	0.79	
	No-injury	0.55	**0.77**	0.66	
XGboost	Death	0.85	0.51	0.63	80
	Injury	0.81	**0.95**	0.88	
	No-injury	0.77	0.43	0.59	
RRVC	Death	**0.95**	**0.64**	**0.76**	**84**
	Injury	0.85	**0.95**	**0.89**	
	No-injury	**0.78**	0.56	0.63	

Table 3
Classification results of ML algorithms using BoW approach without SMOTE (Scenario-1)

Model	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.70	0.40	0.53	73
	Injury	0.76	0.73	0.73	
	No-injury	0.71	0.52	0.61	
Decision tree	Death	0.24	0.39	0.33	71
	Injury	0.75	0.70	0.71	
	No-injury	0.49	0.41	0.46	
Naïve Bayes	Death	0.03	**0.60**	0.32	59
	Injury	**0.94**	0.42	0.65	
	No-injury	0.47	0.68	0.56	
Kernel SVM	Death	0.78	0.12	0.45	71
	Injury	0.84	**0.94**	**0.87**	
	No-injury	**0.72**	0.53	0.62	
Logistic regression	Death	0.13	0.59	0.36	70
	Injury	0.93	0.69	0.79	
	No-injury	0.55	**0.82**	0.66	
XGboost	Death	0.14	0.30	0.28	73
	Injury	0.89	0.72	0.79	
	No-injury	0.54	0.78	0.64	
RRVC	Death	**0.84**	**0.60**	**0.71**	**79**
	Injury	0.80	0.86	0.82	
	No-injury	**0.72**	0.73	**0.71**	

Table 5
Classification results of ML algorithms using TF-IDF approach without SMOTE (Scenario-2)

Model	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.90	0.39	0.59	81
	Injury	0.86	**0.90**	**0.85**	
	No-injury	0.71	0.59	0.61	
Decision tree	Death	0.74	0.47	0.57	73
	Injury	**0.88**	0.86	0.82	
	No-injury	0.63	0.69	0.61	
Naïve Bayes	Death	0.74	0.44	0.55	74
	Injury	0.87	0.89	0.84	
	No-injury	0.69	0.65	0.66	
Kernel SVM	Death	0.37	0.58	0.45	79
	Injury	0.87	0.78	0.82	
	No-injury	0.58	**0.79**	0.64	
Logistic regression	Death	0.33	0.41	0.37	76
	Injury	0.86	0.80	0.81	
	No-injury	0.54	0.63	0.55	
XGboost	Death	0.85	0.51	0.60	80
	Injury	0.81	0.95	0.84	
	No-injury	0.77	0.43	0.58	
RRVC	Death	**0.95**	**0.66**	**0.78**	**85**
	Injury	0.85	**0.90**	**0.85**	
	No-injury	**0.80**	0.62	**0.70**	

Table 4 demonstrates the results of ML algorithms using GloVe as the feature representation method. The performance of the proposed RRVC dropped compared to the TF-IDF feature extraction approach. The performance of other algorithms also dropped, but the DT classifier improved by 2%. The performance of LR was equal to the TF-IDF approach and better than the BoW approach. Similarly, RF performed well and scored 72% accuracy. The XGboost method and RRVC scored the maximum accuracy among all. Additionally, Tables 2, 3 and 4 also illustrate the precision and recall scores of the proposed RRVC classifier, and the results are better than other classifiers. The best results are emphasized in bold. Furthermore, ROC curve is a more appropriate method of calibrating the probability threshold in an imbalanced dataset. Figure 6 depicts the TPR versus FPR in terms of the ROC curve, utilizing IT-IDF as the feature extraction method. Additionally, for further evaluation, the AUC has been incorporated into the ROC for each corresponding category. From Figure 6, it is evident that the maximum AUC is achieved by the proposed RRVC in the categories of Death (89%), Injury (88%), and No-Injury (88%). This demonstrates that the proposed classifier effectively handles the data.

Table 4
Classification results of ML algorithms using GloVe approach without SMOTE (Scenario-1)

Model	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.69	0.48	0.52	72
	Injury	0.72	0.65	0.67	
	No-injury	0.71	0.50	0.61	
Decision tree	Death	0.34	0.31	0.34	71
	Injury	0.67	0.73	0.69	
	No-injury	0.42	0.42	0.42	
Naïve Bayes	Death	0.23	0.42	0.29	62
	Injury	0.67	0.40	0.49	
	No-injury	0.52	0.45	0.47	
Kernel SVM	Death	0.57	0.12	0.29	55
	Injury	**0.84**	**0.94**	**0.89**	
	No-injury	0.54	0.53	0.52	
Logistic regression	Death	0.70	**0.58**	0.59	72
	Injury	0.72	0.71	0.69	
	No-injury	0.42	0.70	0.52	
XGboost	Death	**0.72**	0.55	0.60	**73**
	Injury	0.71	0.75	**0.72**	
	No-injury	0.70	0.42	0.59	
RRVC	Death	**0.72**	0.52	**0.62**	**73**
	Injury	0.73	0.74	**0.72**	
	No-injury	**0.72**	**0.71**	**0.70**	

In conclusion for this scenario, the findings indicate that the GloVe approach outperformed the BoW method, likely attributed to optimal word co-occurrence. However, the TF-IDF approach demonstrated the best results among all three feature representation techniques. The proposed classifier, RRVC, exhibited strong performance in the imbalanced data scenario, achieving the highest accuracy rate when employing TF-IDF as the feature extraction method. Notably, other algorithms, such as XGBoost and RF, also delivered good performance.

3.2.2. Results analysis of scenario 2

In this study, the data instances of the minority class were increased by oversampling in order to balance the classes in the "Reported Event Outcome" column of the dataset. Subsequently, the ML algorithms were trained using the BoW, TF-IDF, and GloVe approaches on the SMOTE-balanced datasets. The results using the TF-IDF approach are presented

Figure 4
Performance analysis of classification model based on accuracy (a) accuracy analysis of classifier without SMOTE and (b) accuracy analysis of classifier with SMO

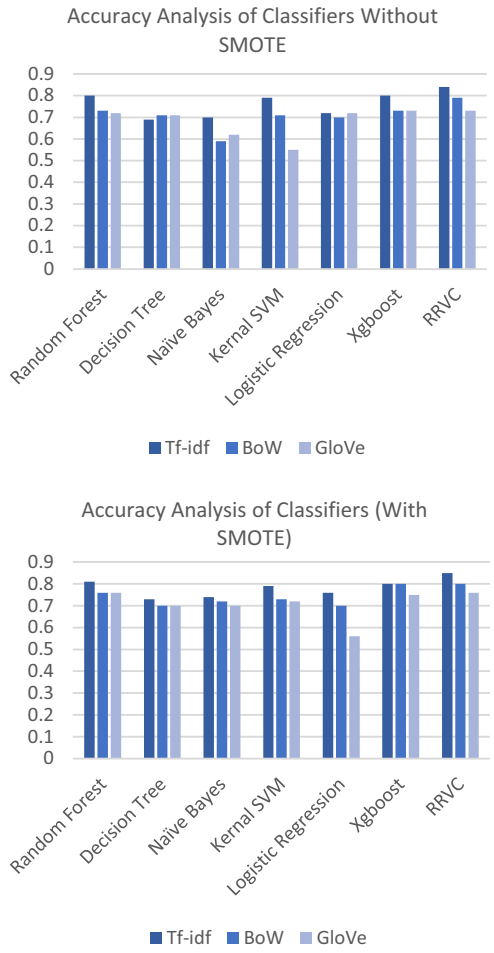
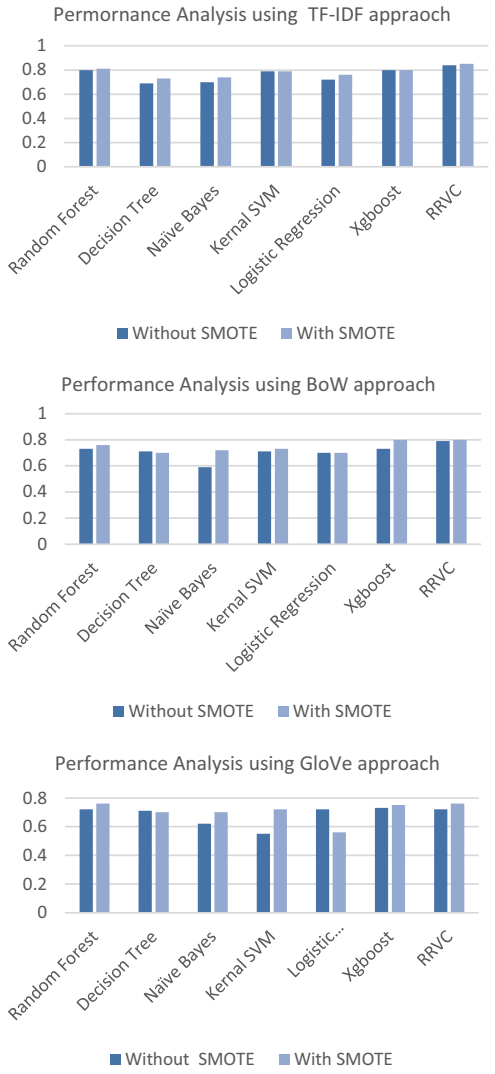


Figure 5
Performance analysis of TF-IDF, BoW, and GloVe with or without SMOTE



in Table 5. It was observed that the results improved significantly, particularly for Naive Bayes and logistic regression. The proposed RRVC classifier also exhibited improved accuracy, but its better results with respect to precision and recall indicate its suitability for the task. The ROC curve is presented in Figure 7, and the AUC attained by the RRVC for instances of Death, Injury, and No-Injury were 92, 89, and 89%, respectively. This indicates that the model is capable of predicting deaths and serious injuries with good accuracy.

In Table 6, satisfactory results are presented using the BoW approach. The proposed classifier and the XGboost classifier achieved the same level of accuracy, while the other algorithms performed well.

Additionally, the results of the ML algorithms using the GloVe approach are presented in Table 7. These results were better than the BoW model, but not as strong as the IT-IDF approach. In all feature extraction methods, the proposed RRVC outperformed the other classifiers and effectively carried out the prediction task. Further analysis of the average F1 score was conducted using Pareto charts for Death and Injury in Figure 8(b) and (d). These two categories of predictions are particularly important, with the death category being of particular significance. A correct prediction in the death category indicates the efficiency of the classifier, as this category had the least representation in the dataset. Figure 8(b) and (d) indicate that the proposed RRVC approach successfully predicted these vulnerable categories.

In conclusion of this scenario, the TF-IDF approach yielded the most significant improvements, with the proposed RRVC classifier consistently outperforming other algorithms in accuracy, precision, and recall. The ROC curve and AUC values confirmed the model's capability, particularly in predicting instances of Death and Injury.

3.2.3. Performance evaluation of ML algorithms using different feature extraction approaches

Here, in this analysis, we will evaluate the performance of different feature extraction techniques. Figure 4(a) illustrates the accuracy comparison of ML classifiers using BoW, TF-IDF, and GloVe without the SMOTE technique, while Figure 4(b) presents the comparison of the same approaches with the SMOTE

Figure 6
ROC curve of ML classifiers (without SMOTE technique) based on TF-IDF

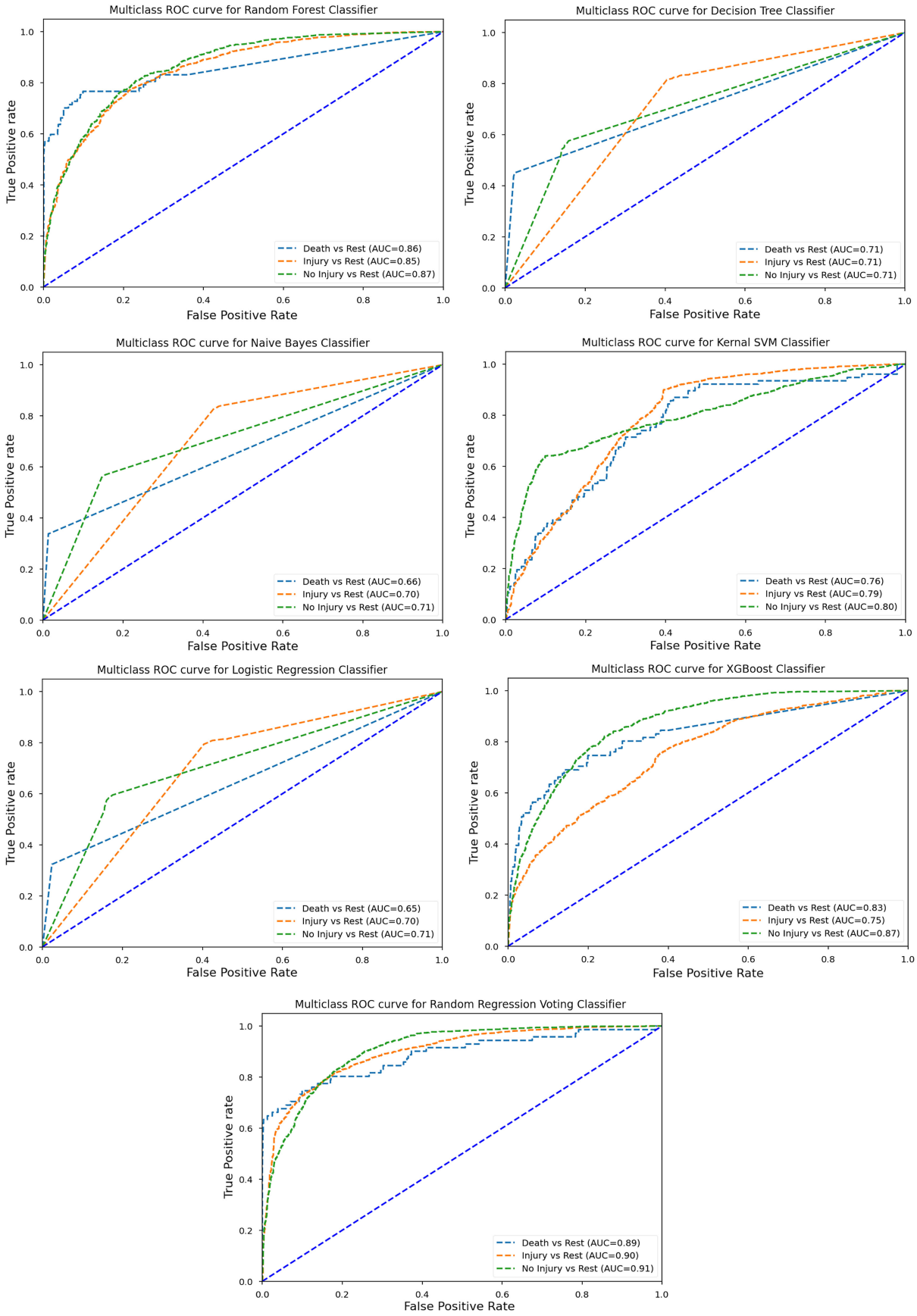


Figure 7
ROC curve of ML classifiers (with SMOTE technique) based on TF-IDF

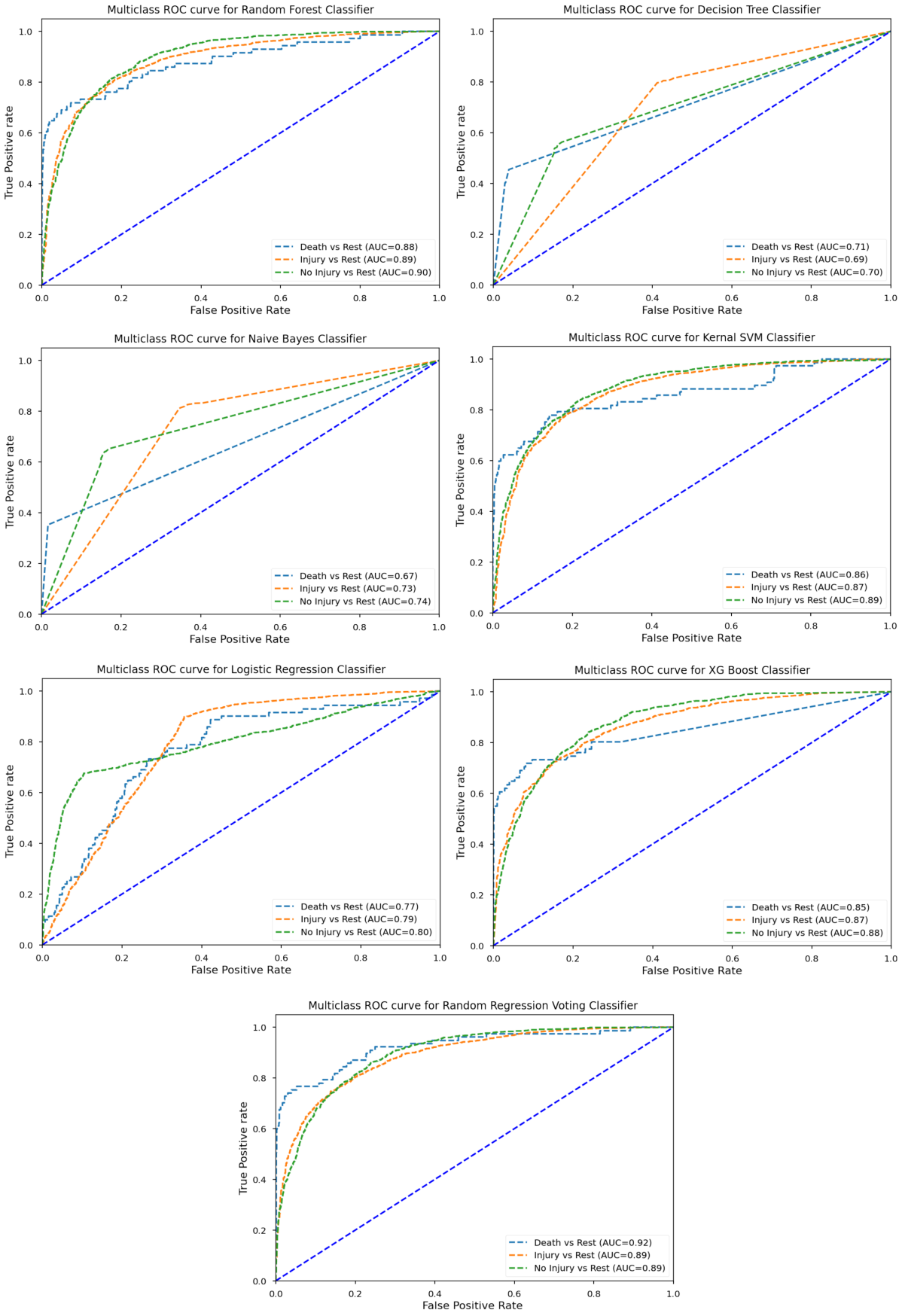
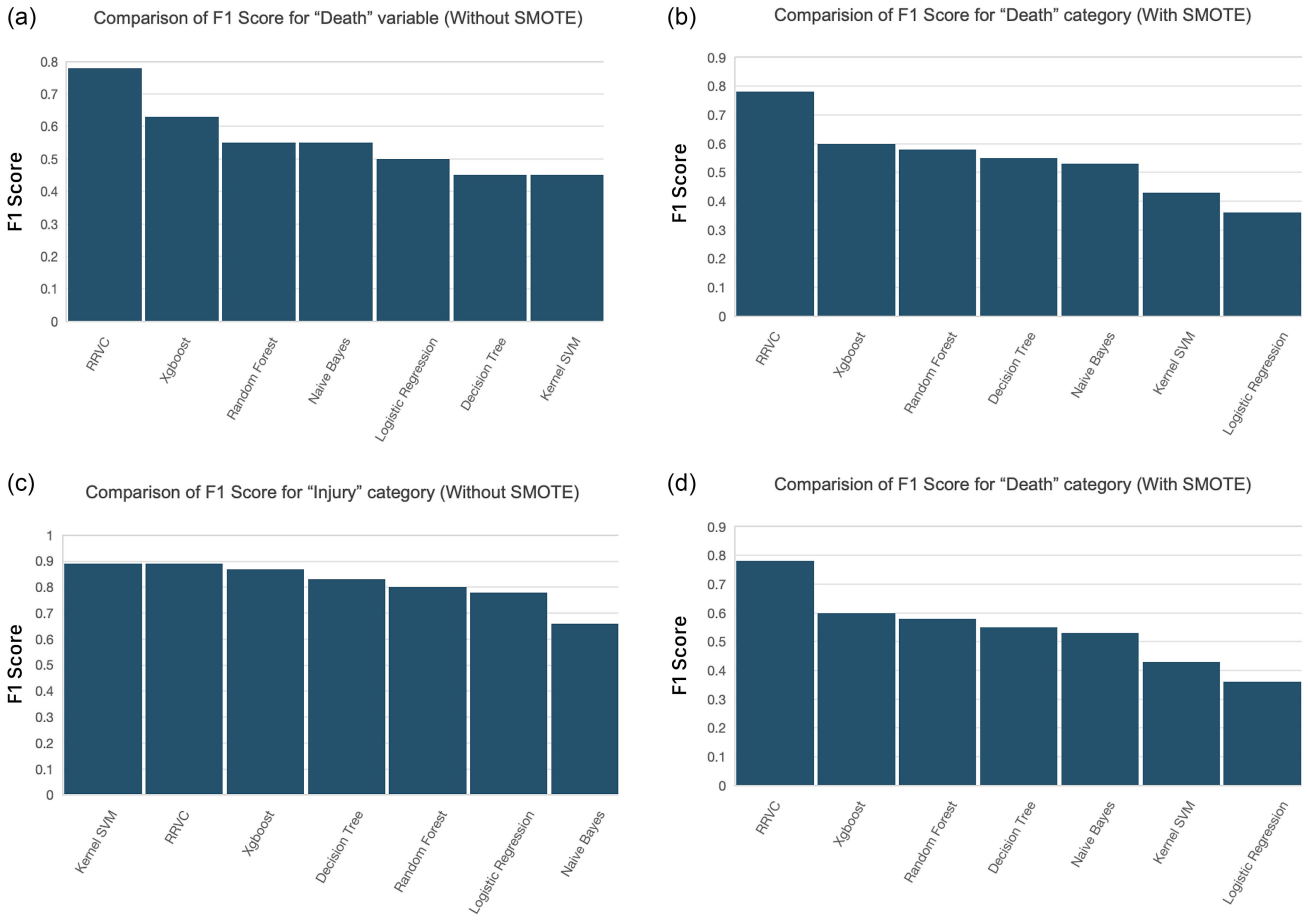


Figure 8
Pareto chart for F1 score (a) death category without SMOTE, (b) death category with SMOTE, (c) injury category without SMOTE and (d) injury category with SMOTE



technique. It can be observed that some classifiers, such as Naive Bayes, LR, and SVM, have substantially improved performance after using SMOTE. However, some classifiers, such as RF, DT, and the proposed RRVC classifier, have shown less improvement. The main reason for this is that the classifier's inbuilt algorithm is strong enough to handle imbalanced datasets, or the construction itself requires fine-tuning, where the user can choose the best possible parameters for the dataset. This requires a deep understanding of the dataset and their parameter behavior. We carefully chose and fine-tuned the parameters, as described in the section on ML classifiers. As a result, the proposed classifier RRVC's performance did not diverge much when we changed the feature extraction.

Figure 5 shows the accuracy comparison of the ML classifier using the TF-IDF, BoW, and GloVe approaches with and without the SMOTE technique. It shows that the results obtained by using the TF-IDF approach with SMOTE are better than the other approaches.

The Pareto chart comparison of the average F1 score for the "Death" and "Injury" categories without SMOTE is shown in Figure 8(a) and (c), while the comparison with SMOTE is shown in Figure 8(b) and (d). In Figures 8(a), (b) and (d), the highest F1 score is achieved by the proposed RRVC approach, which demonstrates its superiority as a predictive model. In Figure 8(c), SVM leads the comparison, with the proposed RRVC following in second place. These results collectively validate the effectiveness of the proposed RRVC model.

The concluding analysis above showcases the effectiveness of the proposed RRVC model in predicting adverse events caused by medical devices. Healthcare practitioners can utilize the model's output to prioritize incidents with a higher risk of adverse events. The model's insights can contribute to proactive measures, leading to a more informed and timely healthcare response. Integrating the model into clinical workflows involves aligning its predictions with existing protocols, ensuring a seamless incorporation of predictive analytics into routine healthcare practices.

While the RRVC classifier shows promise in predicting adverse events related to medical devices, it is important to acknowledge potential limitations. The model's performance could be influenced by the quality and quantity of the training data, making it essential to ensure diverse and representative datasets. Regular

Table 6
Classification results of ML algorithms using BoW approach without SMOTE (Scenario-2)

Model	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.74	0.12	0.49	76
	Injury	0.72	0.81	0.71	
	No-injury	0.70	0.52	0.61	
Decision tree	Death	0.69	0.58	0.59	70
	Injury	0.72	0.70	0.70	
	No-injury	0.53	0.41	0.44	
Naïve Bayes	Death	0.63	0.52	0.46	72
	Injury	**0.84**	0.72	0.74	
	No-injury	0.57	0.65	0.60	
Kernel SVM	Death	0.73	0.12	0.42	73
	Injury	**0.84**	0.80	0.76	
	No-injury	0.69	0.53	0.62	
Logistic regression	Death	0.43	0.49	0.47	70
	Injury	0.83	0.79	0.79	
	No-injury	0.65	0.72	0.56	
XGboost	Death	0.74	0.69	0.67	**80**
	Injury	0.83	0.79	0.78	
	No-injury	**0.77**	**0.62**	**0.64**	
RRVC	Death	**0.75**	**0.70**	**0.70**	**80**
	Injury	0.80	**0.82**	**0.79**	
	No-injury	**0.77**	**0.62**	**0.64**	

Table 7
Classification result of ML algorithms using GloVe approach with SMOTE (Scenario-2)

Model	Parameters	Precision	Recall	F1 score	Accuracy (%)
Random forest	Death	0.70	0.42	0.59	**76**
	Injury	0.69	0.74	0.70	
	No-injury	**0.70**	0.50	**0.62**	
Decision tree	Death	0.52	0.60	0.52	70
	Injury	0.62	0.62	0.62	
	No-injury	0.59	0.50	0.57	
Naïve Bayes	Death	0.49	0.52	0.46	70
	Injury	0.61	0.71	0.59	
	No-injury	**0.70**	0.54	**0.62**	
Kernel SVM	Death	0.55	0.59	0.52	72
	Injury	0.80	**0.82**	0.77	
	No-injury	0.52	0.48	0.49	
Logistic regression	Death	0.43	0.49	0.41	56
	Injury	0.53	0.61	0.59	
	No-injury	0.47	**0.59**	0.51	
XGboost	Death	0.67	0.61	0.62	75
	Injury	0.73	0.74	0.70	
	No-injury	0.55	0.52	0.54	
RRVC	Death	**0.72**	**0.62**	**0.68**	**76**
	Injury	**0.85**	0.76	**0.79**	
	No-injury	0.59	0.58	0.57	

updates and fine-tuning may be necessary to maintain effectiveness as new data becomes available. Ongoing evaluation and validation are crucial to addressing and mitigating these limitations for robust and reliable performance.

4. Conclusion

In conclusion, this study introduces a framework for analyzing adverse events associated with implanted devices. The proposed RRVC model introduced in this study effectively predicts adverse events linked to implanted medical devices by utilizing the DAEN.

The proposed RRVC was evaluated and compared to six other ML algorithms using BoW, TF-IDF, and GloVe. To address data imbalance, experiments were conducted using both the original dataset (after appropriate preprocessing) and a balanced dataset created using the SMOTE technique. Through extensive experiments, it was concluded that the proposed classifier with the SMOTE-balanced dataset and TF-IDF feature extraction showed the best results on this dataset. This study also highlights the potential of ML in improving patient safety and enhancing the effectiveness of medical treatments through better decision making.

Future investigations might delve into advanced feature engineering techniques, real-time prediction capabilities, enhanced interpretability, and cross-domain applicability within healthcare scenarios. This all-encompassing strategy ensures the ongoing evolution of RRVC as a valuable tool for forecasting adverse events and enhancing patient outcomes across diverse medical contexts.

Funding Support

The authors acknowledge the support of the National Health and Medical Research Council (NHMRC) Ideas Grant 2002589.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in the Therapeutic Goods Administration, the Department of Health and Aged Care, Austrilian Government at https://www.tga.gov.au/safety/safety/safety-monitoring-daen-database-adverse-event-notifications/database-adverse-event-notifications-daen.

References

[1] World Health Organization. (2022). *Global atlas of medical devices 2022*. Retrieved from: https://www.who.int/publications/i/item/9789240062207

[2] Shuren, J., & Califf, R. M. (2016). Need for a national evaluation system for health technology. *JAMA*, *316*(11), 1153–1154. https://doi.org/10.1001/jama.2016.8708

[3] Maisel, W. H. (2004). Medical device regulation: An introduction for the practicing physician. *Annals of Internal Medicine*, *140*(4), 296–302. https://doi.org/10.7326/0003-4819-140-4-200402170-00012

[4] Bleu-Laine, M. H., Puranik, T. G., Mavris, D. N., & Matthews, B. (2021). Predicting adverse events and their precursors in aviation using multi-class multiple-instance learning. In *AIAA Scitech Forum*. https://doi.org/10.2514/6.2021-0776

[5] Bala, I. (2023). Natural language processing in medical sciences and healthcare. *Medicon Medical Sciences*, *4*(1), 1–2. https://doi.org/10.55162/MCMS.04.088

[6] Hussain, S., Muhammad, L. J., Ishaq, F. S., Yakubu, A., & Mohammed, I. A. (2019). Performance evaluation of various data mining algorithms on road traffic accident dataset. In *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS, 1,* 67–78. https://doi.org/10.1007/978-981-13-1742-2_7

[7] Jagadeesh, K., & Rajendran, A. (2021). Machine learning approaches for analysis in healthcare informatics. In H. Bansal, B. Balusamy, T. Poongodi & F. K. KP (Eds.), *Machine learning and analytics in healthcare systems: Principles and applications* (pp. 105–122). CRC Press, Boca Raton. https://doi.org/10.1201/9781003185246

[8] Mujtaba, G., Shuib, L., Idris, N., Hoo, W. L., Raj, R. G., Khowaja, K., . . . , & Nweke, H. F. (2019). Clinical text classification research trends: Systematic literature review and open issues. *Expert Systems with Applications*, *116*, 494–520. https://doi.org/10.1016/j.eswa.2018.09.034

[9] Nguyen, H., & Patrick, J. (2016). Text mining in clinical domain: Dealing with noise. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 549–558. https://doi.org/10.1145/2939672.2939720

[10] Zhang, Y., Jiang, M., Wang, J., & Xu, H. (2017). Semantic role labeling of clinical text: Comparing syntactic parsers and features. *AMIA Annual Symposium Proceedings, 2016*, 1283–1292.

[11] Renganathan, V. (2017). Text mining in biomedical domain with emphasis on document clustering. *Healthcare Informatics Research*, *23*(3), 141–146. https://doi.org/10.4258/hir.2017.23.3.141

[12] Turner, C. A., Jacobs, A. D., Marques, C. K., Oates, J. C., Kamen, D. L., Anderson, P. E., & Obeid, J. S. (2017). Word2Vec inversion and traditional text classifiers for phenotyping lupus. *BMC Medical Informatics and Decision Making*, *17*(1), 126. https://doi.org/10.1186/s12911-017-0518-1

[13] Tripoliti, E. E., Papadopoulos, T. G., Karanasiou, G. S., Naka, K. K., & Fotiadis, D. I. (2017). Heart failure: Diagnosis, severity estimation and prediction of adverse events through machine learning techniques. *Computational and Structural Biotechnology Journal*, *15*, 26–47. https://doi.org/10.1016/j.csbj.2016.11.001

[14] Forsyth, A. W., Barzilay, R., Hughes, K. S., Lui, D., Lorenz, K. A., Enzinger, A., . . . , & Lindvall, C. (2018). Machine learning methods to extract documentation of breast cancer symptoms from electronic health records. *Journal of Pain and Symptom Management*, *55*(6), 1492–1499. https://doi.org/10.1016/j.jpainsymman.2018.02.016

[15] Yang, X., Bian, J., Hogan, W. R., & Wu, Y. (2020). Clinical concept extraction using transformers. *Journal of the American Medical Informatics Association*, *27*(12), 1935–1942. https://doi.org/10.1093/jamia/ocaa189

[16] Alsentzer, E., Murphy, J. R., Boag, W., Weng, W. H., Jin, D., Naumann, T., & McDermott, M. (2019). Publicly available clinical BERT embeddings. *arXiv Preprint: 1904.03323*. https://doi.org/10.48550/arXiv.1904.03323

[17] Martenot, V., Masdeu, V., Cupe, J., Gehin, F., Blanchon, M., Dauriat, J., . . . , & Zucker, J. D. (2022). LiSA: An assisted literature search pipeline for detecting serious adverse drug events with deep learning. *BMC Medical Informatics and Decision Making*, *22*(1), 338. https://doi.org/10.1186/s12911-022-02085-0

[18] Therapeutic Goods Administration. (n.d.). *Database of Adverse Event Notifications (DAEN) – Medical devices.* Retrieved from: https://www.tga.gov.au/safety/safety/database-adverse-event-notifications-daen-medical-devices

[19] Gupta, N., & Mathur, P. (2012). Spell checking techniques in NLP: A survey. *International Journal of Advanced Research in Computer Science and Software Engineering*, *2*(12), 217–221.

[20] Manconi, A., Armano, G., Gnocchi, M., & Milanesi, L. (2022). A soft-voting ensemble classifier for detecting patients affected by COVID-19. *Applied Sciences*, *12*(15), 7554. https://doi.org/10.3390/app12157554

[21] Giveki, D. (2021). Scale-space multi-view bag of words for scene categorization. *Multimedia Tools and Applications*, *80*(1), 1223–1245. https://doi.org/10.1007/s11042-020-09759-9

[22] Meijer, H. J., Truong, J., & Karimi, R. (2021). Document embedding for scientific articles: Efficacy of word embeddings vs TFIDF. *arXiv Preprint: 2107.05151*. https://doi.org/10.48550/arXiv.2107.05151

[23] Obayed, H. K., Al-Turaihi, F. S., & Alhussayni, K. H. (2021). Sentiment classification of user's reviews on drugs based on global vectors for word representation and bidirectional long short-term memory recurrent neural network. *Indonesian Journal of Electrical Engineering and Computer Science*, *23*(1), 345–353. http://doi.org/10.11591/ijeecs.v23.i1.pp345-353

[24] Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, *6*(1), 20–29. https://doi.org/10.1145/1007730.1007735

[25] Kadhim, A. I. (2019). Survey on supervised machine learning techniques for automatic text classification. *Artificial Intelligence Review*, *52*(1), 273–292. https://doi.org/10.1007/s10462-018-09677-1

[26] Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST*, *25*(2), 197–227. https://doi.org/10.1007/s11749-016-0481-7

[27] Khanday, A. M. U. D., Rabani, S. T., Khan, Q. R., Rouf, N., & Mohi Ud Din, M. (2020). Machine learning based approaches for detecting COVID-19 using clinical text data. *International Journal of Information Technology*, *12*(3), 731–739. https://doi.org/10.1007/s41870-020-00495-9

[28] Bilal, M., Israr, H., Shahid, M., & Khan, A. (2016). Sentiment classification of Roman-Urdu opinions using Naïve Bayesian, Decision Tree and KNN classification techniques. *Journal of King Saud University-Computer and Information Sciences*, *28*(3), 330–344. https://doi.org/10.1016/j.jksuci.2015.11.003

[29] Kurnia, R., Tangkuman, Y., & Girsang, A. (2020). Classification of user comment using word2vec and SVM classifier. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(1), 643–648.

[30] LaValley, M. P. (2008). Logistic regression. *Circulation*, *117*(18), 2395–2399. https://doi.org/10.1161/CIRCULATIONAHA.106.682658

[31] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

[32] Manning, C., & Schutze, H. (1999). *Foundations of statistical natural language processing*. USA: MIT Press.

[33] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, *34*(1), 1–47. https://doi.org/10.1145/505282.505283

[34] Aguilar-Ruiz, J. S., & Michalak, M. (2022). Multiclass classification performance curve. *IEEE Access*, *10*, 68915–68921. https://doi.org/10.1109/ACCESS.2022.3186444

How to Cite: Bala, I., Kelly, T. L., Lim, R., Gillam, M. H., & Mitchell, L. (2024). An Effective Approach for Multiclass Classification of Adverse Events Using Machine Learning. *Journal of Computational and Cognitive Engineering*, *3*(3), 226–239. https://doi.org/10.47852/bonviewJCCE32021924