Florida International University
FIU Digital Commons

All Faculty

1-1-2021

Predicting Pulsars from Imbalanced Dataset with Hybrid Resampling Approach

Ernesto Lee
Broward College

Furqan Rustam
Khwaja Fareed University of Engineering & Information Technology

Wajdi Aljedaani
University of North Texas

Abid Ishaq
Khwaja Fareed University of Engineering & Information Technology

Vaibhav Rupapara
Florida International University

See next page for additional authors

Follow this and additional works at: https://digitalcommons.fiu.edu/all_faculty

Recommended Citation

Lee, Ernesto; Rustam, Furqan; Aljedaani, Wajdi; Ishaq, Abid; Rupapara, Vaibhav; and Ashraf, Imran, "Predicting Pulsars from Imbalanced Dataset with Hybrid Resampling Approach" (2021). *All Faculty*. 460.
https://digitalcommons.fiu.edu/all_faculty/460

This work is brought to you for free and open access by FIU Digital Commons. It has been accepted for inclusion in All Faculty by an authorized administrator of FIU Digital Commons. For more information, please contact dcc@fiu.edu.

Authors
Ernesto Lee, Furqan Rustam, Wajdi Aljedaani, Abid Ishaq, Vaibhav Rupapara, and Imran Ashraf

This article is available at FIU Digital Commons: https://digitalcommons.fiu.edu/all_faculty/460

Hindawi
Advances in Astronomy
Volume 2021, Article ID 4916494, 13 pages
https://doi.org/10.1155/2021/4916494

Research Article

Predicting Pulsars from Imbalanced Dataset with Hybrid Resampling Approach

Ernesto Lee,[1] **Furqan Rustam**,[2] **Wajdi Aljedaani**,[3] **Abid Ishaq**,[2] **Vaibhav Rupapara**,[4] **and Imran Ashraf**[5]

[1]*Department of Computer Science, Broward College, Broward County, Weston, FL, USA*
[2]*Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan 64200, Pakistan*
[3]*University of North Texas, Denton, TX, USA*
[4]*School of Computing and Information Sciences, Florida International University, Miami, FL, USA*
[5]*Department of Information and Communication Engineering, Yeungnam University, Gyeongsan-si, 38541, Republic of Korea*

Correspondence should be addressed to Vaibhav Rupapara; vaibhav.rupapara.sept@gmail.com and Imran Ashraf; ashrafimran@live.com

Received 22 June 2021; Revised 2 October 2021; Accepted 12 November 2021; Published 3 December 2021

Academic Editor: Jianguo Yan

Copyright © 2021 Ernesto Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Pulsar stars, usually neutron stars, are spherical and compact objects containing a large quantity of mass. Each pulsar star possesses a magnetic field and emits a slightly different pattern of electromagnetic radiation which is used to identify the potential candidates for a real pulsar star. Pulsar stars are considered an important cosmic phenomenon, and scientists use them to study nuclear physics, gravitational waves, and collisions between black holes. Defining the process of automatic detection of pulsar stars can accelerate the study of pulsar stars by scientists. This study contrives an accurate and efficient approach for true pulsar detection using supervised machine learning. For experiments, the high time-resolution (HTRU2) dataset is used in this study. To resolve the data imbalance problem and overcome model overfitting, a hybrid resampling approach is presented in this study. Experiments are performed with imbalanced and balanced datasets using well-known machine learning algorithms. Results demonstrate that the proposed hybrid resampling approach proves highly influential to avoid model overfitting and increase the prediction accuracy. With the proposed hybrid resampling approach, the extra tree classifier achieves a 0.993 accuracy score for true pulsar star prediction.

1. Introduction

Pulsar star represents a stellar remnant often formed by the remains of a collapsed giant star. Usually a neutron star, a pulsar is small in size but contains a large amount of mass. Despite being uncommon, pulsar stars are very important for scientists to study nuclear physics, general relativity, gravitational waves, and factors leading to the collisions of black holes. In 1967, Jocelyn and Anthony Hewish accidentally discovered a pulsar when they were studying distant galaxies [1]. Looking at a particular point through the telescope, they noticed radiation pulses and named them little green men 1 (LGM1). Later these unidentified objects were termed pulsars due to emission as pulses. Now they are called the pulsating source of radiation (PSR), and B1919 + 12 (PSR B1919 + 21) shows the position of the pulsar in the sky [2]. The emission pattern of each pulsar varies over each rotation, so it is averaged over several rotations to determine a star as a pulsar candidate. Without enough radiation, it is very difficult to detect a true pulsar star. However, under certain conditions, detection is possible such as when angled at earth or X-rays burst caused by the detonataion also known as supernova.

Pulsars are the rapidly pivoting astronomical objects detected as a neutron star that emits radiation at the rate of 100,000 km/k to 150,000 km/s with regular intervals and

patterns. Through rays, pulsars emit electromagnetic power that gradually slows down, and pulsars become quiet within ten to a hundred million years. According to the Australian Telescope National Facility (ATNF) catalogue, around 2801 pulsars are identified [3, 4], and an estimated 20,000 to 100,000 pulsars are present in our galaxy indicating that 90% of the pulsars are yet to be identified [5]. Detecting true pulsar is not a trivial task as it is challenging to detect pulsar from the noisy time series data. Each pulsar produces slightly different patterns of signals which make it different from the other signals, and these patterns are called pulsar profiles. In practice, pulsar detection is based on radio frequency interference which makes the identification of legitimate signals very hard. The signals which fulfill the criterion of pulsars are termed as "candidates" and may be termed as new pulsars.

Several automated and human-based methods are used to identify the legitimate candidates for pulsars, and this process is known as "candidate selection" [6]. Until the 2000s, manual selection of candidates was used to find pulsars which generally requires 1–300 s for inspecting each observation [7]. Therefore, for manual inspection of 1 million candidates, up to 80,000 hours of a person are needed. So, manual classification techniques for evaluating pulsar candidates are not appropriate and suitable. Consequently, other techniques are developed to carry out pulsar candidate identification like graphical and automated methods. However, these techniques are computationally expensive as a lot of work is required to uplift the speed and sensitivity of algorithms [8].

By the time, algorithms decreased the ratio of noise in pulsar signals, and signal-to-noise ratio (SNR) became an important factor for pulsar detection. In pulsar astronomy, another important feature called dispersion measure (DM) of the pulsar is also used [9]. The delay of the pulse is associated with DM and radio frequency and has been regarded as an important feature for finding pulsars. Both supervised and unsupervised approaches can be used to perform pulsar detection. For example, unsupervised approaches can be used to group the pulsar data into different clusters whereby the features of each cluster can be further analyzed to select pulsar candidates. This approach is particularly useful for large amounts of unlabeled data. For the HTRU2 dataset, the labels are added by the experts, so supervised machine learning models seem appropriate. One major limitation with the recent works on pulsar detection is the use of imbalanced data. HTRU2 contains a large number of non-pulsar samples while pulsar samples are very few which affects the performance of the classification models. This imbalanced dataset can lead to model overfitting on majority class data. For such models, even though high accuracy is reported, the F1 score is significantly different than the accuracy. Despite the proposal of several automated approaches for finding pulsars, the gap between the provided and the desired accuracy and sensitivity demands further research in this domain. To this end, this study proposes an automated approach for true pulsar prediction using supervised machine learning algorithms and makes the following contributions:

(i) This study devises a methodology for automatic detection of pulsars using the supervised machine learning algorithms. For this purpose, the performance of several well-known machine learning algorithms is analyzed such as random forest (RF), extra tree classifier (ETC), gradient boosting classifier (GBC), and logistic regression (LR). In addition, a multilayer perceptron (MLP) is added in the study.

(ii) The HTRU2 dataset is used for conducting experiments, and the influence of dataset imbalance is extensively investigated. Three resampling approaches such as synthetic minority oversampling technique (SMOTE), adaptive synthetic (ADASYN), and cluster centroids (CC) are studied for their efficacy in data balance. Ultimately, a hybrid data resampling approach, concatenated resampling (CR), is proposed to solve the data imbalance problem of the HTRU2 dataset.

(iii) Extensive experiments are performed to analyze the effect of data balance with SMOTE, ADASYN, CC, and CR on pulsar detection accuracy. Experimental results and performance comparison with state-of-the-art approaches prove that the CR approach performs superior to other resampling approaches.

The rest of the paper is arranged in the following manner. Research papers related to the current study are discussed in Section 2. Section 3 describes the dataset, machine learning algorithms used for experiments, resampling approaches, and the details for the proposed hybrid resampling. Results and discussions are presented in Section 4 while Section 5 provides the conclusion.

2. Related Work

Due to the importance of the detection task for true pulsar stars, several automated approaches have been proposed. These approaches can be broadly categorized under three groups: machine learning approaches, deep learning approaches, and approaches focusing on features' importance. Due to the success of machine learning approaches for various tasks such as classification, object detection, and text analysis, a large number of machine learning-based methods are available in general [10]. However, the pulsar detection domain is not extensively studied and lacks the desired accuracy.

The authors present a machine learning-based approach in [7] for the pulsar selection. It deals with 16 million pulsar candidates obtained from the reprocessing of the Parkes' multibeam survey dataset. A radio transit discovery method named V-FASTR fused random forest is proposed in [11]. V-FASTR has the capacity to consequently shift through realized occasion types with 98.6% accuracy on the training data and 99% on test data. The authors utilize 6 different models to characterize scattered pulsar bunches using signal pulse seek framework in [12]. The dataset used in the research contains 300 pulsars examples and 9600 non-pulsar examples. Several datasets have been generated using

different imbalance treatments. Experimental results show that multiclass ensemble tree learner has high performance and low false positive rate when used with oversampled data.

The study [13] used different machine learning algorithms like GBC, AdaBoost, and XGBoost for the classification of pulsar candidates. To deal with the data imbalance problem, SMOTE is used for oversampling the minority class in the dataset. Several important features from each algorithm are determined for pulsar classification. The major issue with this technique is that the accuracy of radio frequency interference classification is very sensitive to feature selection. The authors present a hybrid machine learning model, random tree boosting voting classifier (RTB-VC), in [14] for pulsar star prediction. RTB-VC combines the free-based classifiers for training on the HTRU2 dataset. RTB-VC uses various combinations of hard voting, soft voting, and weighted voting to obtain high accuracy. A 98.3% F1 score is reported using the proposed RTB-VC model.

Due to the deployment of deep learning approaches in diverse fields for classification and their high accuracy, several deep learning-based models have been adopted for pulsar detection and classification. For example, the authors used a convolutional neural network (CNN) in the PCIS algorithm from the ResNet model for pulsar detection in [15]. On the GBNCC dataset, the proposed system achieved 96% accuracy. Similarly, the research [16] uses an artificial neural network (ANN) for finding true pulsar stars from the HTRU dataset. The research achieves an accuracy of 85% to detect pulsars by visually impaired investigation. It also dismisses 99% of noisy candidates. Both the studies greatly improved recall and decreased the false positive rate. However, the used feature selection method is simple which is based on the hypothesis and subjective to experience. Artificial errors can be made easier which readily affects the performance of the used approaches.

The study [17] focused on pulsar classification using hierarchical deep neural network (DNN). To reduce the training time of DNN, pseudoinverse learning (PIL) is preferred over the gradient descent (GD) method. The proposed model provides 94.65% and 87.66% F1 scores for HTRU medlar and PMPS-26 k datasets, respectively. Despite the low F1 score compared to CNN + BPNN, training time for the proposed model is 5 times low than traditional CNN models. A swift model for the elimination of radio frequency interference (RFI) in pulsar data was proposed in [22]. For learning RFI signatures of real pulsars, PIL-based single hidden layer autoencoder (AE) was used. Results indicate that AE is more robust in learning RFI signatures and can be used to remove them from fast-sampled spectra. As a result, the signals from real pulsars can be obtained. The study [20] investigated the pulsar classification using three datasets: HTRU mid-latitude dataset, the MINIST dataset, and the CIFAR-10 dataset. In the first stage, strong representations for the pulsar candidate are developed in the image domain by extracting deep features with a deep convolutional generative adversarial network (DCGAN). During the second stage, MLP-based classifier is defined using a pseudoinverse learning autoencoder (PILAE). For data imbalance, the SMOTE oversampling technique is used. The achieved accuracy on the HTRU dataset with different data splitting ratios is 100%. On the MINIST dataset, 97.50% accuracy is achieved while CIFAR-10 shows an accuracy of 100%.

The study [6] extracted eight unbiased statistical features including mean, kurtosis, variance, and skewness from the DM curve and pulse profile curve and designed Gaussian–Hellinger fast decision tree for imbalanced data. Using the statistical features on two datasets including HTRU-1 and LOTAAS, 92.8% recall is achieved with a false positive rate of only 0.5%. The research discovered 20 new pulsars from the LOTAAS dataset using the same strategy. A hierarchical candidate shifting model (HCSM) was proposed in [18] where the cost of incorrect prediction of positive samples is emphasized and multiple classifiers are assembled. Handcrafted features are used from three datasets including HTRU, HTRU-1, and LOTAAS to train three classifiers, which collectively make the assemble classifier. Emphasizing the positive examples and assigning higher weights to them produce better results with the proposed model. HCSM achieves a recall value of 97.49% for HTRU dataset, 84.52% for HTRU-1 dataset, and 100% for LOTAAS dataset. A summary of discussed research works is presented in Table 1.

3. Materials and Methods

3.1. Dataset. On account of the importance of pulsar detection, several datasets have been provided for pulsar detection over the years. For the current study, the HTRU2 dataset from Kaggle is used which was collected during high time-resolution [6, 23, 24]. The dataset was compiled by Dr. Robert Lyon and contains 17,998 examples of pulsars and non-pulsars, and 8 features are attributed to each record [6]. The dataset contains examples for pulsar and non-pulsar star observation. The dataset was collected in a high time resolution survey. The survey observed the Galactic plane in the region $-120^{\circ} < l < 30^{\circ}$ and $b \leq 15^{\circ}$. The dataset contains 16,259 specious examples which are the outcomes of RFI/noise, and only 1,639 are pulsar examples; all examples are labeled by human annotators. The dataset does not provide any information related to the pulsars or other astronomical details. Pulsar Feature Lab tool is used to extract pulsar feature data by using candidate files [25]. Table 2 shows the details for the number of samples for pulsar and non-pulsar classes while Table 3 describes the features of the dataset.

3.2. Problem Statement. Keeping in view the results of the related studies discussed in the previous section, it is clear that the dataset used for experiments is not balanced. Similarly, the most commonly used dataset, i.e., the HTRU2 dataset, is highly imbalanced. Only 1,639 samples belong to the pulsar class out of the total 17,998 samples. The class imbalance would result in model overfitting as the machine learning models tend to give higher weight to the class with a higher number of samples. As a result, the F1 score is affected despite good accuracy results from the machine

Table 1: Summary of related work.

Ref	Name	Year	Dataset	Accuracy
[6]	Lyon et al.	2016	HTRU-1 and LOTAAS	Recall = 92.8% and FPS = 50%
[7]	Eatough et al.	2010	650 survey beams of the PMPS	92% of total pulsars are present in dataset of 2.5 million observations
[11]	Wagstaff et al.	2016	ATNF Pulsar Catalogue (not sure)	98.6% accuracy on historical data and a 99%-100% accuracy on (labeled) newly collected data
[12]	Devine et al.	2016	Prestos single_pulse_search on data from the GBT drift-scan (Green Bank Telescope)	
[13]	Bethapudi and Desai	2018	HTRU-S survey data	99.1%
[15]	Wang et al.	2019	PALFA, HTRU, GBNCC, and FAST datasets	96%.
[16]	Bates et al.	2012	HTRU mid-latitude survey	85%
[17]	Li et al.	2018	PMPS-26 k and HTRU dataset	PMPS = 87.50%; HTRU = 97.74%
[18]	Yao et al.	2016	HTRU, HTRU-1, and LOTAAS	Recall: HTRU = 97.49%, HTRU-1 = 84.52%, LOTAAS = 100%
[19]	Mohammed	2018	HTRU-1 dataset	97.8%
[20]	Mohammed and Guo	2020	HTRU, MNIST, and CIFAR-10 datasets	MINIST = 97.50%, HTRU = 100%, CIFAR-10 = 100%.
[21]	Chen et al.	2020	Dataset of pulsar candidate samples collected during high time resolution cosmometry	99%
[14]	Rustum et al.	2020	HTRU2	98.3%

Table 2: Number of samples in HTRU2 dataset.

Dataset	Pulsar	Non-pulsar	Total
HTRU2	1,639	16,259	17,898

Table 3: Description of features in HTRU2 dataset.

Feature description	Data type	Feature
Mean of the integrated profile P	Float	Prof. μ
Standard deviation of the integrated profile P	Float	Prof. σ
Excess kurtosis of the integrated profile P	Float	Prof. k
Skewness of the integrated profile P	Float	Prof.s
Mean of the DM-SNR curve D	Float	DM. μ
Standard deviation of the DM-SNR curve D	Float	DM. σ
Excess kurtosis of the DM-SNR curve D	Float	DM. k
Skewness of the DM-SNR curve D	Float	DM. s
Target class	Binary	

learning models. This study aims at solving this problem by proposing a hybrid resampling approach to achieve high pulsar detection accuracy.

3.3. Data Resampling for Imbalanced Dataset. Looking at the statistics of the dataset given in Table 2, only 1,639 out of 17,898 examples are pulsars while 16,259 are non-pulsars. This is a 1 : 10 ratio which makes the dataset highly imbalanced because the class distribution is skewed towards a specific class. Data imbalance affects the classification performance of the classifiers because the machine learning classifiers tend to the majority class while training. It creates problems for classification. Several approaches can be utilized to deal with the data imbalance. For the present study, two data resampling approaches are adopted.

3.4. Synthetic Minority Oversampling Technique. SMOTE is a widely used oversampling technique to manage imbalanced data [26]. When class distributions are skewed towards a specific class, an imbalanced data problem arises. SMOTE increases the number of data instances by developing random synthetic data of the minority class from its nearest neighbors using Euclidean distance. The newly developed instances are very similar to the original data as the new instances are developed based on the original features [27]. SMOTE is not the best option while dealing with the high-dimensional data because it can create additional noise which is not the case with the HTRU2 dataset used in the current study. SMOTE is adopted based on the results reported in [12, 28] where the data have a ratio of 1 : 10, just like the current study. By generating the samples for the minority class using SMOTE, we get a 1 : 1 ratio of pulsar and non-pulsar as shown in Table 4. For SMOTE implementation, we used an open-source Python toolbox, called imbalanced-learn which uses Scikit-learn, SciPy, and NumPy.

3.5. Adaptive Synthetic Resampling. ADASYN is used for upsampling the minority class samples in an imbalanced dataset [29, 30]. Being the enhanced form of SMOTE,

Table 4: HTRU2 dataset count after applying SMOTE resampling.

	Original	SMOTE
Total	17,898	32,518
Non-pulsars	16,259	16,259
Pulsar	1,639	16,259

ADASYN has been regarded as superior to SMOTE. ADASYN generates synthetic alternatives for observations of the minority class. The ease and difficulty of generating observations depend upon the learning difficulty. An observation is "hard to learn" if several observations exist in the majority class, having similar features to that of minority class observation. It essentially leads to the observation surrounded by majority class instances when plotted in the features space which makes it harder for the models to learn. Due to its efficiency and reliability, it is widely used in many applications like detection of cancer, credit card fraud detection, and so on.

3.6. Cluster Centroids. Besides using SMOTE and ADASYN oversampling approaches, this study utilizes cluster centroid undersampling approaches to downsize the majority class. During this process, clusters of the majority class are formed and the whole cluster is replaced with the centroid to undersample it. For this purpose, the current study uses the K-mean algorithm to find the clusters of the majority class.

3.7. Supervised Machine Learning Models. For performing classification, several types of machine learning models are available. The availability of open-source library Scikit-learn helps researchers to solve classification problems using machine learning and ensemble learning [31]. Well-known machine learning algorithms are selected due to their reported performance. Instead of devising new models, already established models are selected, and their performance is optimized using several hyperparameters. The machine learning models used in this research are RF, LR, GBC, ETC, and MLP. Several parameters of these models are fine-tuned to optimize the performance, and the list of used parameters is provided in Table 5.

3.8. Random Forest. RF is a tree-based ensemble learning model, which produces accurate predictions by combining many weak learners [32]. The bagging technique is used where a variety of decision trees are used during training with various bootstrap samples [33]. A bootstrap sample is derived by subsampling the training dataset with replacement, where the size of the sample is the same as that of the training dataset. RF uses decision trees for the prediction process, and a big issue in the construction of decision trees is proof of identity of the attributes for root nodes at each level. This method is termed attribute selection. In ensemble classification, some classifiers are trained and their results are pooled through a voting process. Previously, many researchers have proposed ensemble learning approaches [34–36]. The widely used ensemble learning methods are bagging [37] and boosting [38, 39]. In the bagging (or bootstrap aggregating) technique, classifiers are trained on the bootstrap samples to minimize the variance of classification. RF has the following mathematical form:

$$p = \text{mode}T1(y), T2(y), \ldots, Tm(y), \tag{1}$$

$$p = \text{mode}\sum_{m=1}^{M} Tm(y), \tag{2}$$

where p is the final prediction by the majority of decision trees and $T1(y), T2(y), \ldots, Tm(y)$ is the number of decision trees taking part in the production process.

3.9. Gradient Boosting Classifier. In GBC, several weak learning classifiers work together to create a strong learning model. The working principle of gradient boosting is time-consuming and computationally expensive because it creates several independent trees. Gradient boosting has been previously used by several studies in astronomy [24]. For example, study [40] uses GBC for photometric classification of supernova while study [41] uses GBC for the detection and classification of galaxy using Galaxy Zoo catalogue. Mean square error (MSE) is used in the GBC as

$$L = \text{MSE} = \sum\left(h_i - h_i^p\right)^2, \tag{3}$$

where L is the loss, h_i is the ith target value, h_i^p shows the ith prediction, and (h_i, h_i^p) is the loss function.

Based on learning rate, GBC updates predictions and finds the values where MSE has the minimum value. Minimization of MSE is represented using the equation below:

$$h_i^p = \frac{h_i^p + r * \delta\sum\left(h_i - h_i^p\right)^2}{\delta h_i^p}, \tag{4}$$

$$h_i^p = h_i^p - r * 2 * \sum\left(h_i - h_i^p\right)^2, \tag{5}$$

where r is the learning rate and $(h_i + h_i^p)$ is the sum of all the residual values which are near to 0 or minimum and predicted values are very close to the actual values.

3.10. Extra Tree Classifier. ETC is a meta-estimator also known as the extra randomized tree that uses extra decision trees and fits them into various subsamples of the dataset. To improve the accuracy, it uses the averaging technique and controls the overfitting of the model. ETC works similar to RF, but the difference lies in the construction of trees in the forest. In ETC, each tree is made from the original training sample. Random samples of K best features are used for decision and the Gini index is used to select the top feature to separate the data in the tree. ETC has been utilized to perform various tasks in astronomy. For example, study [42] uses the ETC model for neutrinos detection from a point-like source with the collaboration of KM3Net which is the cubic kilometer neutrino telescope.

Table 5: Hyperparameters for machine learning models.

Algorithms	Parameters
RF	n_estimators = 200, max_depth = 8 random_state = 2
GBC	n_estimators = 100, max_depth = 10, random_state = 2, learning_rate = 0.1
ETC	n_estimator = 200, max_depth = 8, random_state = 2
LR	random_state = 2, solver = liblinear, multi_class = ovr, C = 3.0

3.11. Logistic Regression. LR is a statistical method used to deal with classification problems. LR analyzes the data to estimate the probability of class members. For classification problems where the target variables are categorical, LR is the first choice to perform classification. It processes the relationship between categorical dependent variables and one or more independent variables by estimating probability using the logistic function. A logistic curve or logistic function is a common "S" shaped or sigmoid curve and is defined as

$$f(x) = \frac{L}{1 + e^{-m(v - v_o)}}, \tag{6}$$

where e is the Euler number, v_o is the x-value of the sigmoid midpoint, L is the curve's maximum value, and m shows the steepness of the curve. LR works well on binary classification and shows good performance for text classification as well [43, 44].

3.12. Multilayer Perceptron. An MLP consists of one or more layers of neurons. MLP is a feed-forward neural network model which maps the set of input data to a set of appropriate outputs and every layer is fully connected. Data are fed into the input layer that passes through one or more hidden layers. The hidden layers provide the level of abstraction, and predictions are made on the visible or output layer [45]. Multiple neurons can be stacked in one layer, and multiple layers have better predictive capacity.

The MLP model consists of three layers: one input layer, one hidden layer, and one output layer. We used 32 neurons in the input layer with ReLU activation function, 64 neurons in the hidden layer, and the output layer used one neuron with a sigmoid activation function. The value used for the dropout layer is 0.2. For compilation, we used Adam optimizer, binary_crossentropy loss function, and 100 epochs.

3.13. Proposed Resampling Approach. This study proposes a data resampling approach called combined resampling (CR). CR is a resampling technique that concatenates the results of three resampling techniques including SMOTE, ADASYN, and CC for enhancing the prediction results. Results of all three resampling techniques are concatenated along the horizontal axis, which increases the size of the data. CR is defined as

$$\mathrm{HTRU2}_{(\mathrm{pulsar,non-pulsar})} = \text{pulsar and non} - \text{pulsar examples}. \tag{7}$$

Here HTRU2 is the original dataset with an imbalanced target class ratio.

$$\mathrm{ADASYN}_{(i,j)} = \mathrm{ADASYN}\left(\mathrm{HTRU2}_{(\mathrm{pulsar,non-pulsar})}\right), \tag{8}$$

$$\mathrm{SMOTE}_{(u,v)} = \mathrm{SMOTE}\left(\mathrm{HTRU2}_{(\mathrm{pulsar,non-pulsar})}\right), \tag{9}$$

$$\mathrm{CC}_{(p,q)} = \mathrm{CC}\left(\mathrm{HTRU2}_{(\mathrm{pulsar,non-pulsar})}\right). \tag{10}$$

$\mathrm{ADASYN}_{(i,j)}$ refers to the output data after balancing target ratio using the ADASYN technique; similarly, $\mathrm{SMOTE}_{(u,v)}$ and $\mathrm{CC}_{(p,q)}$ are data outputs after SMOTE and CC are applied on the original HTRU2 dataset, while u, p, i represent the number of features/attributes and i, q, j represent the number of records.

$$\mathrm{CR}_{(m,n)} = \left[\mathrm{ADASYN}_{(i,j)}, \mathrm{SMOTE}_{(u,v)}, \mathrm{CC}_{(p,q)}\right]. \tag{11}$$

Here, $\mathrm{ADASYN}(I, j)$, $\mathrm{SMOTE}(u, v)$, and $\mathrm{CC}(p, q)$ are the results of the ADASYN, SMOTE, and CC techniques, respectively, while $\mathrm{CR}(m, n)$ is the concatenation result of these three resampling techniques. Additionally, $m = i = u = p$ shows the number of attributes, and $n = j + v + q$ is the number of records. Figure 1 illustrates the proposed CR approach to perform resampling from the original dataset, where RS1, RS3, and RS3 are resampled instances of data from three different techniques which are combined to make the new sampled dataset.

3.14. Proposed Methodology for Pulsar Detection. For detecting the pulsars, the current study leverages the supervised machine learning approach. The concept of ensemble and hybrid approaches is very popular in the machine learning task. A number of studies can be found that leverage hybrid and ensemble models for a variety of tasks in several domains such as image processing, classification, text analysis, and so on [46, 47]. For example, study [48] uses a stack generalization technique and ensemble learning approach for pulsar prediction. Similarly, ensemble approaches are also used for predicting the numeric scores for Google apps in [49]. Hybrid or ensemble approaches are also used for text analysis [50]. Results reported for hybrid approaches provide the motivation to utilize a hybrid approach for the task at hand.

The flow of the proposed methodology is shown in Figure 2. As the first step, the HTRU2 dataset is obtained from Kaggle. The HTRU2 dataset contains pulsar and non-pulsar examples in an unequal ratio with non-pulsar examples as majority class and pulsar as a minority class. Owing to the influence of data imbalance on the performance of the classifiers, this problem is solved using the proposed approach. For analyzing the influence of data splitting on the prediction accuracy, data splitting is

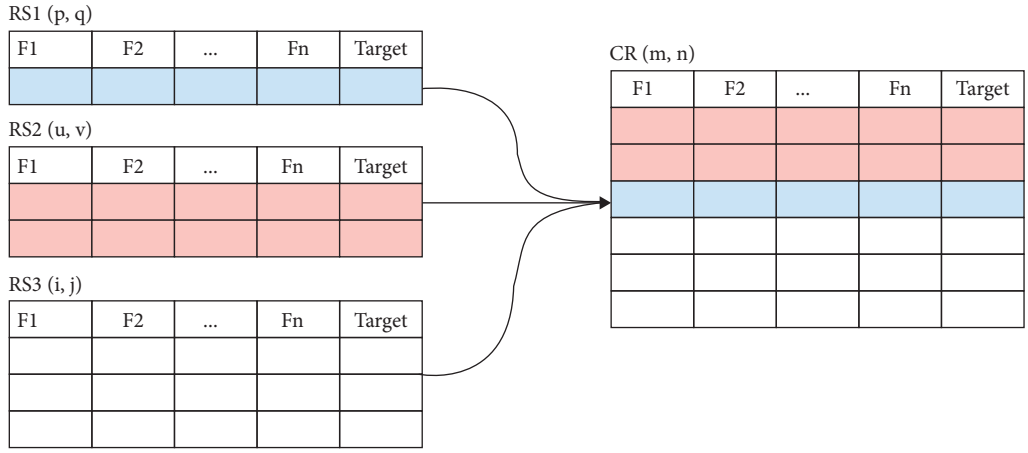

Figure 1: Proposed resampling approach diagram.

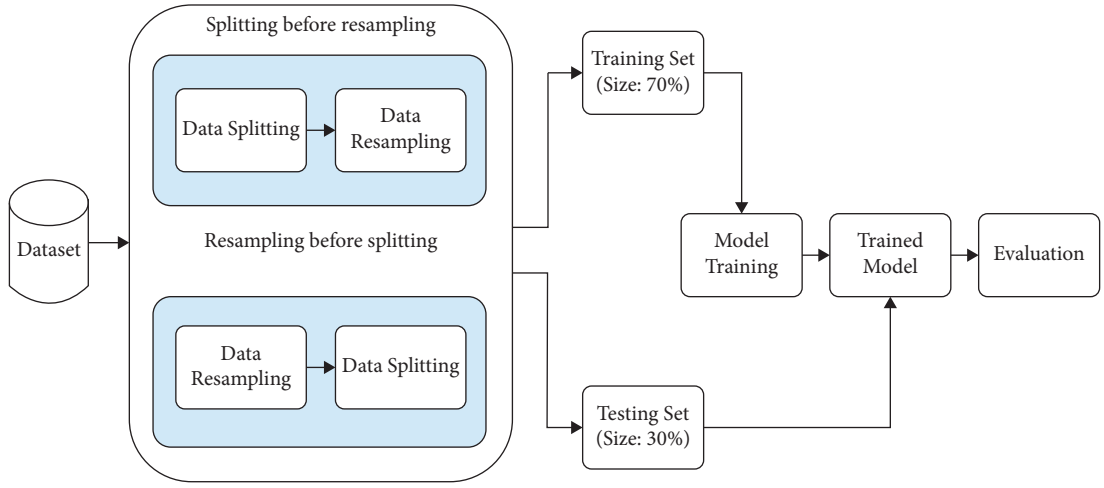

Figure 2: The flow of the proposed methodology.

performed before resampling and resampling before splitting, in a ratio of 70 : 30 for both approaches. When data are split before resampling, resampling is applied only on the training set. For data balancing, CC, SMOTE, ADASYN, and CR techniques are used. Table 6 shows the count for both pulsar and non-pulsar samples when resampling is performed before splitting, and Table 7 shows the count for both pulsar and non-pulsar samples when resampling is applied only on the training set.

After data splitting and resampling, machine learning models are trained including RF, ETC, GBC, LR, and MLP using 70% of data. The rest (30%) is used to evaluate the trained models. The evaluation is performed using accuracy, precision, recall, and F1 score.

3.15. Performance Evaluation Metrics. Several performance evaluation methods are used to evaluate the machine learning models. The blend of different evaluation tools is helpful to determine the efficacy of an approach [51]. Therefore, in this research, four well-known metrics are used including accuracy, precision, recall, and F1 score. In addition, the confusion matrix helps to show true positive (TP), true negative (TN), false positive (FP), and false negative (FN) which are used to calculate the values for accuracy, precision, recall, and F1 score. These metrics are calculated using the following equations:

Table 6: Number of pulsar and non-pulsar samples after data resampling before data splitting.

Category	Original	CC	SMOTE	ADASYN	CR
Pulsar	1,639	1,639	16,259	16,103	34,001
Non-pulsar	16,259	1,639	16,259	16,259	34,157
Total	17,898	3,278	32,518	32,362	68,158

Table 7: Number of samples for pulsar and non-pulsar samples for data resampling after data splitting.

Category	Original	CC	SMOTE	ADASYN	CR
Pulsar	1,639	1,639	13,321	13,331	28,291
Non-pulsar	16,259	4,577	16,259	16,259	37,095
Total	17,898	6,216	29,580	29,590	65,386

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{12}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{13}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{14}$$

$$\text{F1 score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{15}$$

4. Results and Discussion

This study performs experiments using a Core i7 7th generation machine operating on Windows 10. Implementation of the machine learning algorithms is done using Python script on Jupyter Notebook.

4.1. Results without Resampling. The performance of machine learning models without data resampling is shown in Table 8. The performance of RF is the highest as compared to other models with 0.980 and 0.887 scores for accuracy and F1 score, respectively. Performance of LR is marginally low with 0.980 accuracy and 0.885 F1 score. A noteworthy point is a difference in the prediction accuracy and F1 score. Such difference in the accuracy and F1 score is often caused by the data imbalance. Models have an overfit due to high number of samples in the majority class and make false predictions for the minority class, leading to the difference in the prediction accuracy and F1 score.

4.2. Results Using CC Undersampling. To improve the performance of machine learning models, data resampling is carried out using the CC technique. The CC technique is used for data balancing and reduces the chances of the model overfitting. The CC technique is an undersampling approach that reduces the number of samples of the majority class by randomly selecting the records and removing them, thus making the number of samples of the majority and minority class equal.

Results given in Table 9 indicate that the difference in the prediction accuracy and F1 score has been reduced after applying the resampling. Using an equal number of samples for training reduces the probability of model overfitting and reduces the gap between accuracy and other performance evaluation metrics. On the other hand, the overall performance of the machine learning models is reduced as well. The primary reason for this downfall in performance is the size of the data used for models' training. Being a data undersampling approach, CC reduces the size of data, and models' training is affected which leads to performance degradation. Despite a decrease in the performance of different models, RF shows the best performance with the undersampled data and achieves 0.943 accuracy score and 0.940 F1 score. The performance of other classifiers is similar except for MLP which achieves an accuracy of 0.905 and F1 score of 0.898.

Table 8: Results of machine learning models without data resampling.

Model	Accuracy	Precision	Recall	F1 score
RF	0.980	0.925	0.859	0.887
ETC	0.979	0.931	0.834	0.880
GBC	0.978	0.912	0.851	0.881
LR	0.980	0.942	0.831	0.885
MLP	0.975	0.902	0.826	0.862

Table 9: Results of machine learning models with CC data resampling.

Model	Accuracy	Precision	Recall	F1 score
RF	0.943	0.958	0.922	0.940
ETC	0.935	0.949	0.915	0.932
GBC	0.936	0.946	0.922	0.934
LR	0.933	0.961	0.899	0.929
MLP	0.905	0.938	0.861	0.898

4.3. Results Using SMOTE Oversampling. The performance of machine learning models after data oversampling is shown in Table 10. Results indicate that the performance of the machine learning models has been elevated when trained on the oversampled data using SMOTE. Oversampling increases the size of data which provides large feature set to train the models which boost their prediction accuracy. As for the performance of the machine learning models, ETC outperforms all models with an accuracy of 0.982 and F1 score of 0.982. All other models also show improvement in their performance with SMOTE oversampling technique. The performance of RF is slightly lower than that of ETC with an accuracy of 0.976. Overall, tree-based models show prominent performance as compared to linear and neural network models. Tree-based models perform significantly better due to their ensemble architecture. ETC, RF, and GBC combine several decision trees in learning and prediction procedures and perform superior on the HTRU2 dataset.

4.4. Results after Applying ADASYN Sampling. For the current study, ADASYN oversampling is also used to balance the dataset. The performance of machine learning models using the ADASYN oversampled data is shown in Table 11. Results suggest that the performance of the machine learning models has been improved when used with ADASYN oversampled data. Tree-based models again outperform linear models and MLP and achieve good scores for performance evaluation metrics. For example, ETC achieves the highest accuracy score of 0.981 and F1 score of 0.982. The performance of linear model LR and neural network model MLP dropped when used with ADASYN resampling because of the dataset's new sample feature correlation.

4.5. Results with Proposed Combined Resampling. For the proposed approach, resampled data from SMOTE, ADASYN, and CC are concatenated along the 0 axis which increases the size of data and leads to significant improvement in the performance of machine learning models. Results

Table 10: Results of machine learning models with SMOTE data resampling.

Model	Accuracy	Precision	Recall	F1 score
RF	0.976	0.983	0.969	0.976
ETC	0.982	0.986	0.979	0.982
GBC	0.966	0.978	0.955	0.966
LR	0.943	0.973	0.912	0.941
MLP	0.927	0.945	0.908	0.926

Table 11: Results of machine learning models with ADASYN data resampling.

Model	Accuracy	Precision	Recall	F1 score
RF	0.973	0.965	0.982	0.974
ETC	0.981	0.974	0.989	0.982
GBC	0.961	0.949	0.974	0.961
LR	0.839	0.871	0.795	0.832
MLP	0.860	0.858	0.861	0.860

shown in Table 12 indicate that machine learning models perform better with the proposed CR sampling approach. Both ETC and RF achieve >99% accuracy with the CR technique with a similar F1 score which indicates that the models do not experience overfitting when trained with CR resampled data. The elevated performance is due to the concatenation of resampled data from different sampling approaches. It provided the models with different variations of samples to learn and make them more significant as compared to an individual data resampling technique. As a result, the performance of machine learning models has been significantly improved.

Using the proposed feature resampling approach, ETC outperforms with all resampling techniques and most significantly with the proposed CR resampling approach as shown in Figure 3.

Figure 4 shows the confusion matrix of the best performer ETC with all resampling approaches. The confusion matrix shows that ETC makes 20,327 correct predictions out of 20,448 total predictions with only 121 false predictions with CR resampling. On the other hand, when ETC is used with SMOTE, 166 predictions are false and 9,590 predictions are correct out of 9,756 total predictions. Out of the 166 total false predictions, the model makes 101 false predictions from the resampled data which indicates the data generated by SMOTE to balance the dataset lead to false predictions. For the ADASYN case, the ETC model performs slightly poor than SMOTE as it makes 9,543 correct and 166 false predictions out of 9,709 total predictions. In the case of the CC undersampling technique, the performance is not good enough due to the reduced number of samples used for training. ETC gives 921 correct and 63 false predictions out of 984 total predictions. In light of discussed results, the performance of machine learning models when used with the proposed CR resampling approach is better than that of both oversampling and undersampling approaches.

4.6. Results with Resampling on Training Set. Due to the highest performance of ETC with all the resampling approaches used for the current study, ETC is used for further analysis. For this purpose, the training dataset is balanced and ETC is trained on the balanced dataset while tested on the imbalanced dataset. Results given in Table 13 show that ETC outperforms all other models with this approach as well. ETC achieves the highest accuracy of 0.981 with the proposed CR resampling approach. However, the overall performance of the model has been reduced following this approach. Furthermore, values for accuracy and F1 score are sharply different than the values obtained in the previous approach.

Table 12: Results of machine learning models with CR data resampling.

Model	Accuracy	Precision	Recall	F1 score
RF	0.991	0.992	0.990	0.991
ETC	0.993	0.992	0.994	0.993
GBC	0.971	0.968	0.974	0.971
LR	0.886	0.924	0.840	0.880
MLP	0.892	0.910	0.868	0.889

4.7. Results with Deep Learning Models. This study also deploys the state-of-the-art deep learning models for pulsar detection. Customized architectures of long short-term memory (LSTM), deep neural network (DNN) [10], and gated recurrent unit (GRU) models are used [52]. Architectural details and list and values of used variables are provided in Table 14.

Deep learning models are compiled with binary cross-entropy and Adam optimizer, and 100 epochs are used for training. The performance of LSTM, GRU, and DNN is measured in terms of accuracy, precision, recall, and F1 score. Performance results given in Table 15 indicate that the achieved accuracy from three deep learning models is the same. However, the performance has marginal variance when precision, recall, and F1 scores are considered. Owing to the importance of the F1 score, LSTM and GRU show a better F1 score of 0.94 each as compared to the DNN model. Results prove that the optimized machine learning models have superior performance than deep learning models. Model fitting for deep learning models requires thousands of samples to show better performance; consequently, their performance is slightly less than that of machine learning models due to the small size of the dataset.

4.8. Results Using 10-Fold Cross-Validation. To corroborate the significance of the proposed resampling approach and performance of the machine learning models, 10-fold cross-validation is used, and results are given in Table 16. All models are employed with each data sampling approach to analyze the performance. Results indicate that the highest accuracy is obtained by ETC with the proposed hybrid sampling approach which shows the supremacy of the proposed approach over other data sampling approaches.

4.9. Comparison with the State-of-the-Art Studies. For evaluating the efficacy of the proposed approach, a performance comparison is done with the previous similar

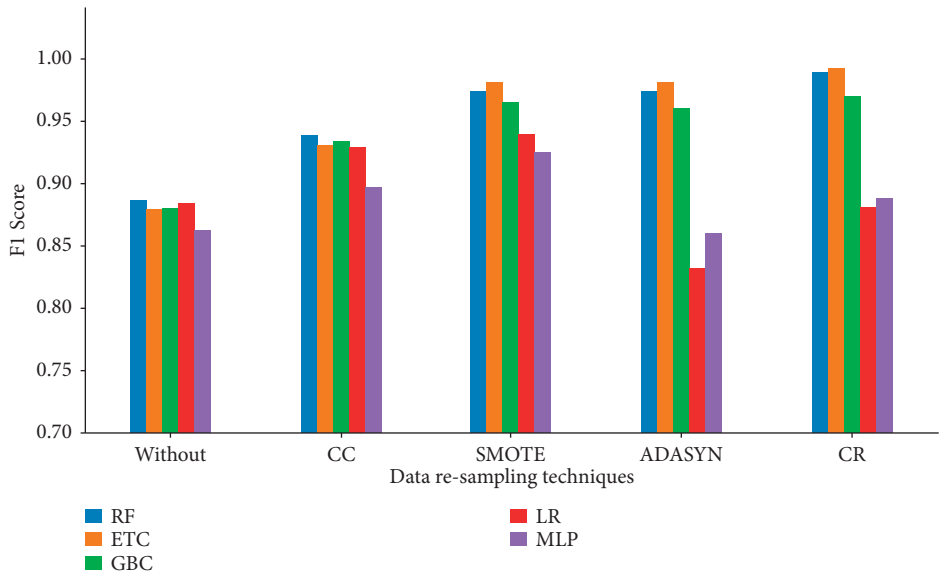

Figure 3: Comparison of F1 score for all approaches.

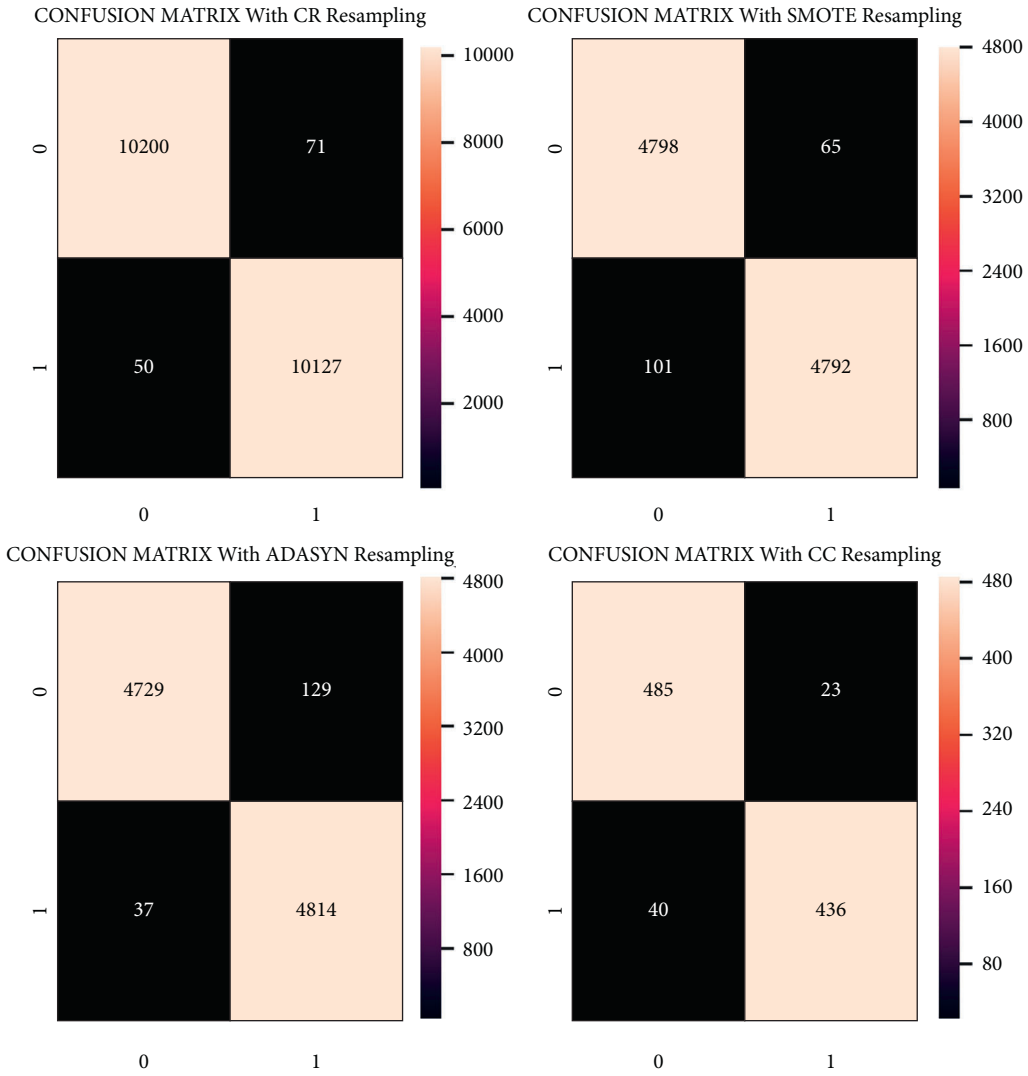

Figure 4: Confusion matrix of best performer ETC using all resampling techniques.

Table 13: Results of best performer ETC using all resampling techniques when data splitting was performed before data resampling.

Resampling	Accuracy	Precision	Recall	F1 score
CR	0.981	0.887	0.894	0.892
SMOTE	0.978	0.877	0.889	0.883
ADASYN	0.972	0.811	0.906	0.856
CC	0.959	0.720	0.914	0.805

Table 14: Architecture of LSTM, DNN, and GRU used for experiments.

LSTM	DNN	GRU
LSTM (32)	Dense (64, activation = "relu")	GRU (64, return_sequences = True))
Dropout (0.2)	Dropout (0.2)	SimpleRNN (32)
Dense (64, activation = "relu")	Dense (64, activation = "relu")	Dense (32)
Dropout (0.2)	Dropout (0.2)	Dropout (0.2)
Dense (2, activation = "softmax")	Dense (2, activation = "softmax")	Dense (16)
—	—	Dense (2, activation = "softmax")
Loss = "binary_crossentropy," optimizer = "Adam," epochs = 100		

Table 15: Performance results for deep learning models.

Model	Accuracy	Precision	Recall	F1 Score
LSTM	0.98	0.96	0.91	0.94
DNN	0.98	0.96	0.90	0.93
GRU	0.98	0.95	0.92	0.94

Table 16: Results of 10-fold cross-validation.

Models	CR	SMOTE	ADASYN	CC
RF	0.98 (± 0.01)	0.97 (± 0.01)	0.96 (± 0.01)	0.94 (± 0.02)
ETC	0.99 (± 0.00)	0.98 (± 0.00)	0.98 (± 0.01)	0.94 (± 0.02)
GBC	0.98 (± 0.00)	0.97 (± 0.01)	0.97 (± 0.02)	0.94 (± 0.02)
LR	0.98 (± 0.01)	0.96 (± 0.01)	0.96 (± 0.04)	0.93 (± 0.02)
MLP	0.97 (± 0.01)	0.93 (± 0.01)	0.93 (± 0.03)	0.91 (± 0.02)

approaches. To this end, approaches that utilize the HTRU2 dataset have been selected. For example, study [6] conducted experiments using the same dataset with the proposed GH-VFDT model. Similarly, study [14] performed experiments on the same dataset using the proposed RTB-VC for pulsar prediction. Table 17 shows the comparison of the proposed approach with the previous studies to illustrate the significance of the study.

4.10. Statistical Analysis of CR Technique. This study also performs a statistical *T*-test to show the significance of CR techniques. The *T*-test considers two hypotheses as follows:

(i) Null hypothesis: the CR technique is statistically significant to other data balancing techniques.

(ii) Alternative hypothesis: the CR technique is not statistically significant to other data balancing techniques.

The *T*-test shows that the results of tree-based models RF, ETC, and GBC with CR techniques accept the null hypothesis and reject the alternative hypothesis which means that these tree-based models are statistically significant with the CR technique as compared to all other resampling techniques.

Table 17: Performance comparison with previous studies.

Study	Model	Year	Accuracy	F1 score
[6]	GH-VFDT	2017	0.978	0.862
[14]	RTB-VC	2020	0.983	0.982
[48]	Ensemble	2021	0.980	—
[53]	LR	2021	0.970	0.870
[54]	RF	2021	0.980	0.880
Current study	ETC	2021	0.992	0.993

5. Conclusion

Pulsar detection is a significant task and possesses great importance for studying several phenomena of nuclear physics. Automatic detection of pulsars from the collected data is a topic of significant importance in this regard. Due to the imbalanced nature of the HTRU2 dataset, the prediction accuracy is not up to the standard. This study proposes a concatenated resampling (CR) approach for data balance and a methodology to utilize the proposed CR for pulsar prediction with high accuracy. For this purpose, the performance of several machine learning algorithms is investigated and analyzed. Experimental results indicate that oversampling approaches SMOTE and ADASYN perform better than the undersampling cluster centroid approach. The increased feature vector for the oversampled data tends to boost the performance of the machine learning classifiers, especially the ETC, which achieves the highest accuracy with all resampling approaches. Performance evaluation metrics are much better for ETC when used with the proposed CR

approach with an accuracy of 0.993. Combining multiple resampling approaches elevates the performance of machine learning classifiers and reduces the influence of data imbalance. Results show that tree-based classifiers perform better than linear classifiers. Regarding the use of deep learning models, LSTM and GRU provide better F1 scores than DNN. Performance comparison with state-of-the-art approaches indicates that the proposed approach outperforms them and achieves higher accuracy.

This study leverages the supervised approach by optimizing several well-known machine learning models. However, the use of unsupervised models is expected to provide interesting results. Important observations can be made by clustering the HTRU dataset into groups, and analysis can be performed to highlight the features of probable candidates for pulsars.

Data Availability

The HTRU2 dataset is available at https://www.kaggle.com/colearninglounge/predicting-pulsar-starintermediate.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Authors' Contributions

Ernesto Lee and Furqan Rustam contributed equally to this study.

Acknowledgments

This research was supported by the Florida Center for Advanced Analytics and Data Science funded by Ernesto.-Net (under the Algorithms for Good Grant).

References

[1] H. Anthony, M. Elder, P. Davies, and J. Bell-Burnell, *The Discovery of the Pulsar*, Audio-Forum, New York, NY, USA, 1981.

[2] A. A. Ershov and A. D. Kuzmin, "Detection of giant pulses in pulsar psr j1752+ 2359," *Astronomy & Astrophysics*, vol. 443, no. 2, pp. 593–597, 2005.

[3] "Atnf pulsar catalogue," 2019, https://www.atnf.csiro.au/research/pulsar/psrcat/.

[4] R. N. Manchester, G. B. Hobbs, A. Teoh, and M. Hobbs, "The Australia telescope national facility pulsar catalogue," *The Astronomical Journal*, vol. 129, no. 4, 2005.

[5] S. Johnston and A. Karastergiou, "Pulsar braking and the p–diagram," *Monthly Notices of the Royal Astronomical Society*, vol. 467, no. 3, pp. 3493–3499, 2017.

[6] R. J. Lyon, B. W. Stappers, S. Cooper, J. M. Brooke, and J. D. Knowles, "Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach," *Monthly Notices of the Royal Astronomical Society*, vol. 459, no. 1, pp. 1104–1123, 2016.

[7] R. P. Eatough, N. Molkenthin, M. Kramer et al., "Selection of radio pulsar candidates using artificial neural networks," *Monthly Notices of the Royal Astronomical Society*, vol. 407, no. 4, pp. 2443–2450, 2010.

[8] M. J. Keith, R. P. Eatough, A. G. Lyne et al., "Discovery of 28 pulsars using new techniques for sorting pulsar candidates," *Monthly Notices of the Royal Astronomical Society*, vol. 395, no. 2, pp. 837–846, 2009.

[9] M. A. McLaughlin, A. G. Lyne, D. R. Lorimer et al., "Transient radio bursts from rotating neutron stars," *Nature*, vol. 439, no. 7078, pp. 817–820, 2006.

[10] F. Rustam, A. A. Reshi, I. Ashraf et al., "Sensor-based human activity recognition using deep stacked multilayered perceptron model," *IEEE Access*, vol. 8, pp. 218898–218910, 2020.

[11] K. L. Wagstaff, B. Tang, D. R. Thompson et al., "A machine learning classifier for fast radio burst detection at the vlba," *Publications of the Astronomical Society of the Pacific*, vol. 128, no. 966, Article ID 084503, 2016.

[12] T. R. Devine, K. Goseva-Popstojanova, and M. McLaughlin, "Detection of dispersed radio pulses: a machine learning approach to candidate identification and classification," *Monthly Notices of the Royal Astronomical Society*, vol. 459, no. 2, pp. 1519–1532, 2016.

[13] S. Bethapudi and S. Desai, "Separation of pulsar signals from noise using supervised machine learning algorithms," *Astronomy and computing*, vol. 23, no. 15–26, 2018.

[14] F. Rustam, A. Mehmood, S. Ullah et al., "Predicting pulsar stars using a random tree boosting voting classifier (rtb-vc)," *Astronomy and Computing*, vol. 32, Article ID 100404, 2020.

[15] H. F. Wang, W. W. Zhu, P. Guo et al., "Pulsar candidate selection using ensemble networks for fast drift-scan survey," *Science China Physics, Mechanics & Astronomy*, vol. 62, no. 5, pp. 1–10, 2019.

[16] S. D. Bates, M. Bailes, B. R. Barsdell et al., "The high time resolution universe pulsar survey—vi. an artificial neural network and timing of 75 pulsars," *Monthly Notices of the Royal Astronomical Society*, vol. 427, no. 2, pp. 1052–1065, 2012.

[17] S. Li, S. Feng, P. Guo, and Q. Yin, "A hierarchical model with pseudoinverse learning algorithm optimazation for pulsar candidate selection," in *Proceedings of the 2018 IEEE Congress on Evolutionary Computation (CEC)*, July 2018.

[18] Y. Yao, X. Xin, and P. Guo, "Pulsar candidate selection by assembling positive sample emphasized classifiers," in *Proceedings of the 2016 12th International Conference on Computational Intelligence and Security (CIS)*, December 2016.

[19] T. M. Mohamed, "Pulsar selection using fuzzy knn classifier," *Future Computing and Informatics Journal*, vol. 3, no. 1, pp. 1–6, 2018.

[20] M. A. B. Mahmoud and P. Guo, "Learning from imbalanced pulsar data by combine dcgan and pilae algorithm," *New Astronomy*, vol. 85, Article ID 101561, 2021.

[21] Z. Chen, A. Xu, Y. Zhou, and Y. Gai, "Research on pulsar classification based on machine learning," in *Proceedings of the 2020 3rd International Conference on Intelligent Autonomous Systems (ICoIAS)*, February 2020.

[22] H.-F. Wang, M. Yuan, Q. Yin et al., "Radio frequency interference mitigation using pseudoinverse learning autoencoders," *Research in Astronomy and Astrophysics*, vol. 20, no. 8, 2020.

[23] "HTRU2 dataset," 2021, https://www.kaggle.com/colearninglounge/predicting-pulsar-starintermediate/.

[24] S. Bethapudi and S. Desai, "Separation of pulsar signals from noise with supervised machine learning algorithms," 2017, https://arxiv.org/abs/1704.04659.

[25] "Pulsar feature lab tool," 2019, https://figshare.com/articles/PulsarFeatureLab/1536472/1/.

[26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[27] N. V. Chawla, *Data Mining and Knowledge Discovery Handbook*, Springer, Berlin, Germany, 2009pp. 875–886, Data mining for imbalanced datasets: An overview.

[28] R. Barandela, R. M. Valdovinos, J. S. Sánchez, and F. J. Ferri, "The imbalanced training sample problem: under or over sampling?" in *Proceedings of the Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*, pp. 806–814, Springer, Lisbon, Portugal, August 2004.

[29] H. He, B. Yang, E. A. Garcia, and S. Li, "Adasyn: adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, IEEE, Hong Kong, China, June 2008.

[30] A. Aditsania, Adiwijaya, and A. L. Saonard, "Handling imbalanced data in churn prediction using adasyn and backpropagation algorithm," in *Proceedings of the 2017 3rd international conference on science in information technology (ICSITech)*, pp. 533–536, IEEE, Bandung, Indonesia, October 2017.

[31] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: machine learning in python," *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[32] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: a classification and regression tool for compound classification and qsar modeling," *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[33] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[34] J. A. Benediktsson and P. H. Swain, "Consensus theoretic classification methods," *IEEE transactions on Systems, Man, and Cybernetics*, vol. 22, no. 4, pp. 688–704, 1992.

[35] L. K. Hansen and P. Salamon, "Neural network ensembles," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 10, pp. 993–1001, 1990.

[36] L. I. Kuncheva, "That elusive diversity in classifier ensembles," in *Proceedings of the Iberian Conference on Pattern Recognition and Image Analysis*, pp. 1126–1138, Springer, Mallorca, Spain, June 2003.

[37] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[38] R. E. Schapire, "A brief introduction to boosting," in *Ijcai*vol. 99, , pp. 1401–1406, Citeseer, 1999.

[39] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[40] A. Möller, V. Ruhlmann-Kleider, C. Leloup et al., "Photometric classification of type ia supernovae in the supernova legacy survey with supervised learning," *Journal of Cosmology and Astroparticle Physics*, vol. 2016, no. 008, 2016.

[41] A. Tramacere, D. Paraficz, P. Dubath, J.-P. Kneib, and F. Courbin, "Asterism: application of topometric clustering algorithms in automatic galaxy detection and classification," *Monthly Notices of the Royal Astronomical Society*, vol. 463, no. 3, pp. 2939–2957, 2016.

[42] A. Trovato, R. Coniglione, P. Sapienza, and J. Barrios-Martí, "Expectations for detection of neutrinos from point-like sources with km3net/arca," in *Proceedings of the 35th International Cosmic Ray Conference—PoS (ICRC2017). 35th International Cosmic Ray Conference*, vol. 301, p. 999, Busan, Korea, July 2017.

[43] E. Guzman and W. Maalej, "How do users like this feature? a fine grained sentiment analysis of app reviews," in *Proceedings of the 2014 IEEE 22nd International Requirements Engineering conference (RE)*, pp. 153–162, IEEE, Karlskrona, Sweden, August 2014.

[44] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[45] M. Banerjee, S. Mitra, and S. K. Pal, "Rough fuzzy mlp: knowledge encoding and classification," *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1203–1216, 1998.

[46] A. Imran, S. Hur, and Y. Park, "Application of deep convolutional neural networks and smartphone sensors for indoor localization," *Applied Sciences*, vol. 9, no. 11, p. 2337, 2019.

[47] M. Khalid, I. Ashraf, A. Mehmood, S. Ullah, M. Ahmad, and G. S. Choi, "Gbsvm: sentiment classification from unstructured reviews using ensemble classifier," *Applied Sciences*, vol. 10, no. 8, p. 2788, 2020.

[48] M. Murali, H. Sharma, and A. Mishra, "Pulsar star detection using ensemble algorithms and stack generalization," in *Proceedings of the 6th International Conference on Recent Trends in Computing*, pp. 153–169, Springer, Delhi, India, April 2021.

[49] M. Umer, A. Imran, A. Mehmood, S. Ullah, and G. S. Choi, "Predicting numeric ratings for google apps using text features and ensemble learning," *ETRI Journal*, vol. 43, no. 1, pp. 95–108, 2021.

[50] A. Mehmood, On Byung-Won, I. Lee, A. Imran, and G. S. Choi, "Spam comments prediction using stacking with ensemble learning," in *Journal of Physics: Conference Series*vol. 933, IOP Publishing, Article ID 012012, 2017.

[51] L. Yang, Y. Zheng, X. Cai et al., "A lstm based model for personalized context-aware citation recommendation," *IEEE access*, vol. 6, pp. 59618–59627, 2018.

[52] R. Jamil, I. Ashraf, F. Rustam, E. Saad, A. Mehmood, and G. S. Choi, "Detecting sarcasm in multi-domain datasets using convolutional neural networks and long short term memory network model," *PeerJ Computer Science*, vol. 7, p. e645, 2021.

[53] D. Beniwal, A. Roy, H. Yadav, and A. Chauhan, "Detection of pulsars by classical machine learning algorithms," in *Proceedings of the 2021 2nd International Conference for Emerging Technology (INCET)*, May 2021.

[54] J. Holewik and G. Schaefer, "Ensemble classifiers for pulsar detection," *Intelligent Astrophysics*, vol. 39, pp. 153–166, 2021.