IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 May 2025, accepted 22 June 2025, date of publication 24 June 2025, date of current version 1 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3582762

RESEARCH ARTICLE

Toward Automatic Detection of Pi2 Magnetic Pulsation Using Machine Learning

MOHAMED S. ABDALZAHER[1,2], (Senior Member, IEEE), ESSAM GHAMRY[3], KHAIRUL ADIB YUSOF[4], AND MOSTAFA SHAABAN[1,5], (Senior Member, IEEE)

[1]Department of Electrical Engineering, American University of Sharjah, Sharjah 26666, United Arab Emirates
[2]Department of Seismology, National Research Institute of Astronomy and Geophysics, Helwan, Cairo 11421, Egypt
[3]Department of Geomagnetic and Geoelectric, National Research Institute of Astronomy and Geophysics, Helwan, Cairo 11421, Egypt
[4]Department of Physics, Faculty of Science, Universiti Putra Malaysia, Serdang 43400, Malaysia
[5]Energy, Water, and Sustainable Environment Research Center, College of Engineering, American University of Sharjah, Sharjah 26666, United Arab Emirates

Corresponding author: Mohamed S. Abdalzaher (msalah@aus.edu; msabdalzaher@nriag.sci.eg)

The work in this paper was supported, in part, by the open access program and project FRG24-C-E66 from the American University of Sharjah. This paper represents the opinions of the author(s) and does not mean to represent the position or opinions of the American University of Sharjah.

ABSTRACT Magnetic pulsations of type Pi2 are a well-established category of Ultra Low Frequency (ULF) waves, characterized by irregularly damped oscillations with periods ranging from 40 to 150 seconds (6.7–25 mHz). Nowadays, it is well known that Pi2 occurs at the onset of geomagnetic substorms, is considered an outstanding research topic in space physics, and is a link between ionospheric and magnetospheric processes. The discontinuation of the conventional index previously employed to detect Pi2 pulsations has driven this study to propose an innovative detection method leveraging machine learning (ML). This paper introduces a novel ML-based classification framework that utilizes geomagnetic field data for Pi2 pulsation detection. A comprehensive analysis of various linear, ensemble, and non-linear ML models was conducted, employing hyperparameter optimization to identify the optimal model with high classification performance and minimal computational overhead during testing. Model robustness was assessed using multiple evaluation metrics, including accuracy, F1-score, kappa score, execution time, precision-recall curves, ROC curves, learning curves, and confusion matrices. The proposed gradient boost (GB) classifier demonstrated superior performance, achieving 98.21% accuracy in distinguishing Pi2 pulsations. This detection system offers a reliable and efficient tool for monitoring Pi2 pulsations in the nighttime, contributing to advancements in space weather analysis and substorm detection.

INDEX TERMS Machine learning, Pi2 pulsations, geomagnetic field, automatic detection, geophysical data classification.

I. INTRODUCTION

One of the most interesting space weather phenomena is the geomagnetic substorm, which is considered a complex phenomenon due to enhanced plasma convection from the tail towards the Earth [1]. It lasts a few hours and can be divided into three distinct phases: a growth phase, which lasts typically about an hour and usually occurs when IMF has a southward component, in which energy is delivered from the solar wind to be stored in the night side tail of the magnetosphere; an expansion stage, lasts typically few minutes. Accordingly, the stored energy is explosively released as a recovery phase that usually takes several hours, and the magnetosphere relaxes to a quiet state [2], [3]. Their occurrence triggers various phenomena, including auroral breakups, the release of energetic particles into the magnetosphere, magnetic field dipolarization, explosive plasma flows into the plasma sheet, and magnetic reconnection within the magnetotail. Substorms also disrupt the geomagnetic field at the Earth's surface, especially in high and mid-latitude regions [2]. Despite their relatively small scale, substorms have significant impacts on Earth and its inhabitants, affecting telecommunications, navigation, and electric power distribution systems [4]. Therefore, early detection of substorms is crucial. One potential early indicator of substorms is Pi2 pulsations. The Pi2 pulsations

The associate editor coordinating the review of this manuscript and approving it for publication was Geng-Ming Jiang.

are an irregular waveform of Ultra Low Frequency (ULF) waves. These waves are commonly detected as geomagnetic field variations with a period of 40–150 s [5]. Interestingly, many researchers have studied this type of pulsation observed on the ground and in space, and can be selected by visual inspection [6]. It is well known that Pi2 pulsations indicate the onset of magnetospheric substorms and are visible clearly in mid- or low-latitudes on the nightside [7], [8], [9], [10], [11]. At the same time, several studies show that Pi2 can occur during non-substorm intervals [12], [13].

Since 2005, Pi2 pulsation can be detected using the W_p index developed by the Institute for Space-Earth Environmental Research through the Substorm Swift Search project [14]. However, this project ceased publishing the index data for public use at the end of 2019, creating a need for alternative methods to detect Pi2 pulsations.

The authors in [15] and [16] investigated the statistical characteristics of Pi2 and utilized wavelet analysis for the automatic detection of Pi2 pulsations. Although there is widespread wavelet usage, the wavelet method can be computationally intensive and sensitive to the choice of wavelet basis, scaling levels, and boundary conditions, which may lead to overfitting. It may be less intuitive, less effective for stationary signals, and challenging to apply to irregular data. Therefore, a more reliable, robust, and effective technique is still desired.

Recently, ML has proved beneficial for the accurate detection of many applications, especially the ones of time series datesets [17], [18], [19], [20], [21], [22], [23], [24]. However, deep learning has proved beneficial in the literature [25], [26], [27], [28], [29], ML excels deep learning due to the low complexity of implementation. The application of ML to detect Pi2 pulsations presents a promising area for exploration. However, using ML for this purpose is not an easy task, especially when dealing with time series data. In recent years, advancements in ML have led to the automation of routine ML model development procedures, a process known as automated machine learning (AutoML). An AutoML framework includes three steps: feature engineering, hyperparameter tuning, and algorithm selection. Then, the three steps are performed iteratively and assisted by an optimizer [30]. Using AutoML reduces the time by testing multiple algorithms and suitable hyperparameter combinations to obtain the best model without human intervention [31], [32], [33].

A. MOTIVATION

It is well known that Pi2 pulsations are seen as reliable indicators of substorms, which are considered crucial for understanding the coupling between the solar wind, magnetosphere, and ionosphere. The prediction of substorms is vital for advancing space weather forecasting because it could lead to disturbances for satellites, communication systems, power grids, and surface charging, which is one of the most common causes of spacecraft anomalies [4], [34], [35].

Traditional methods for predicting Pi2 pulsations are limited by their inability to effectively handle the high-dimensional, noisy, and non-linear nature of solar-terrestrial interactions. Accordingly, the main motivation to present this study is the need for early detection of geomagnetic substorms via Pi2 pulsation detection, the discontinuation of the W_p index publication, previously used as an indicator of Pi2. It is worth mentioning that the W_p index uses only geomagnetic stations on the nightside (1800–0400 MLT). Besides, this index can be considered "a global index" reflecting Pi2 wave power during local nighttime at any given UT time [14]. Consequently, such a phenomenon requires a reliable, advanced, and robust solution such as ML.

Introducing ML to predict Pi2 pulsations represents a novel approach, addressing these limitations by leveraging advanced algorithms to model the intricate dynamics of the system. ML provides a transformative solution by utilizing large, high-quality datasets from space missions and ground-based observatories. Unlike conventional methods, ML models can uncover hidden relationships, analyze complex non-linear systems, and adapt to evolving data, significantly improving accuracy and real-time prediction capabilities. This innovative application of ML enhances operational space weather forecasting and sets a precedent for leveraging AI in geophysical research. Accordingly, this paper enhances our understanding of Earth's near-space environment while mitigating the societal risks of space weather events.

B. CONTRIBUTION

The main contributions of the paper are as follows:

- Introducing an ML-based classification approach for classifying the Pi2 pulsation, thereby enhancing reliable and controllable satellite, communication, and smart grid systems against the effects of such events.
- Comparing several linear, ensemble, and non-linear models and determining the best model based on the highest accuracy and minimal execution time for classifying the Pi2 pulsation.
- Achieving exceptional accuracy rates of 98.21% with the developed GB model, thus outperforming benchmark models and confirming its reliability in Pi2 pulsation detection.

The structure of the paper is presented as follows. The data preparation for the developed model is presented in Section II. The proposed model, formulations, and system architecture are described in Section III. Then, the utilized evaluation metrics are discussed in Section IV. Afterward, the obtained results are portrayed and discussed in Section V. Finally, the paper is concluded in Section VI.

II. DATA PREPARATION

Two types of data were utilized in this study: the W_p index and the geomagnetic field. The W_p index is related to wave power of low-latitude Pi2 pulsations [14], and derived based

TABLE 1. Details of the stations used for derivation of the W_p index.

Location	Code	Geographic Latitude (°)	Geographic Longitude (°)
Tucson	TUC	32.2	249.3
Honolulu	HON	21.3	202.0
Canberra	CNB	-35.3	149.4
Kakioka	KAK	36.2	140.2
Learmonth	LRM	-22.2	114.1
Urumqi	WMQ	43.8	87.7
Iznik	IZN	40.5	29.7
Fürstenfeldbruck	FUR	48.2	11.3
Ebro	EBR	40.8	0.5
Tristan da Cunha	TDC	-37.3	347.5
San Juan	SJG	18.1	293.9

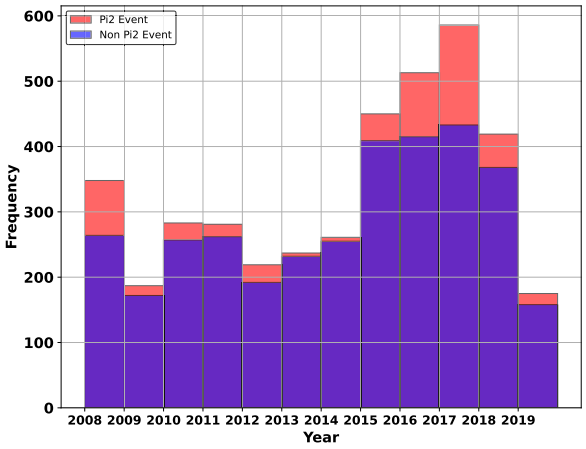

FIGURE 1. Pi2 and Non-Pi2 distribution.

on magnetometer data from 11 stations located at low and mid-latitudes as shown in Table 1.

In this study, the W_p index data was obtained from the online database at [36]. The data was limited to the project's duration, from the 1st January 2005 to 31st December 2019. The index values range between 0 and 10, with over 99% of the data points falling within values less than 0.5. Consequently, this study adopted a threshold value of $W_p > 0.5$ to identify 18202 Pi2 pulse events during the period when the index was available (2008 — 2019) as illustrated in Figure 1.

For the geomagnetic field, 1 Hz frequency data was obtained from the magnetometer network INTERMAGNET, which provides open-access data [37]. This data includes three components: northward (N), eastward (E), and downward (Z). For each Pi2 pulsation, a 10-minute geomagnetic dataset was generated, encompassing the horizontal component (H) from each index station (Table 1) that was in local nighttime (between 22:00 and 02:00). The presence of Pi2 signals in the geomagnetic data was verified using the following algorithm: (i) filtering the data within the frequency range of 6.7 to 25.0×10^{-3} Hz, (ii) detecting signal peaks within the first 150 seconds that reached a minimum amplitude of 1 nT, (iii) if detection was successful, the Pi2 pulse was considered present, and the dataset was classified as Pi2. If peak detection failed, it was categorized as Non-Pi2. It is worth mentioning that this index is derived from data collected by multiple stations (11 geomagnetic stations at low latitudes) distributed around the globe at different longitudes and always located on the nightside, where low-latitude Pi2 pulsations can be clearly observed. It is calculated by applying bandpass filtering and power spectral density estimation within sliding time windows. A commonly used threshold (Wp > 0.5) is employed to identify Pi2 events [14], [38], [39].

These two classes were necessary to achieve the study's objective of binary classification. A total of 3959 Pi2 datasets were successfully generated, while the number of Non-Pi2 datasets was smaller. Figure 2 shows a sample for the Pi2 and Non-Pi2 signals. To augment the Non-Pi2 data, the observation window was shifted 20 seconds forward or backward until a 10-minute dataset during $W_p \leq 0.5$ was found. This search was restricted to the same local nighttime period to minimize diurnal or seasonal influences. Despite the augmentation, only 3417 Non-Pi2 datasets could be generated.

III. SYSTEM MODEL AND DEVELOPMENT MACHINE LEARNING APPROACHES

This section presents the utilized process to classify the Pi2 pulsation using ML. In this regard, we examined several linear, ensemble, and non-linear ML models to pinpoint the best model fitting the nature of geomagnetic data with the highest accuracy and minimum elapsed time for the Pi2 pulsation classification. Figure 3 shows an overview of the proposed method starting with the data collection till reaching the best model of classification. The flow of classification is executed through **Algorithm** 1.

A. LINEAR ML MODELS

1) LOGISTIC REGRESSION

This type is a fundamental classifier in the ML field [40], [41]. LR estimates the likelihood that a given input is related to a specific label using the logistic or sigmoid function, to the linear combination of input features. This results in an output $\in [0, 1]$. This output clarifies the probability of the input being classified as the positive target (denoted by 1), while the

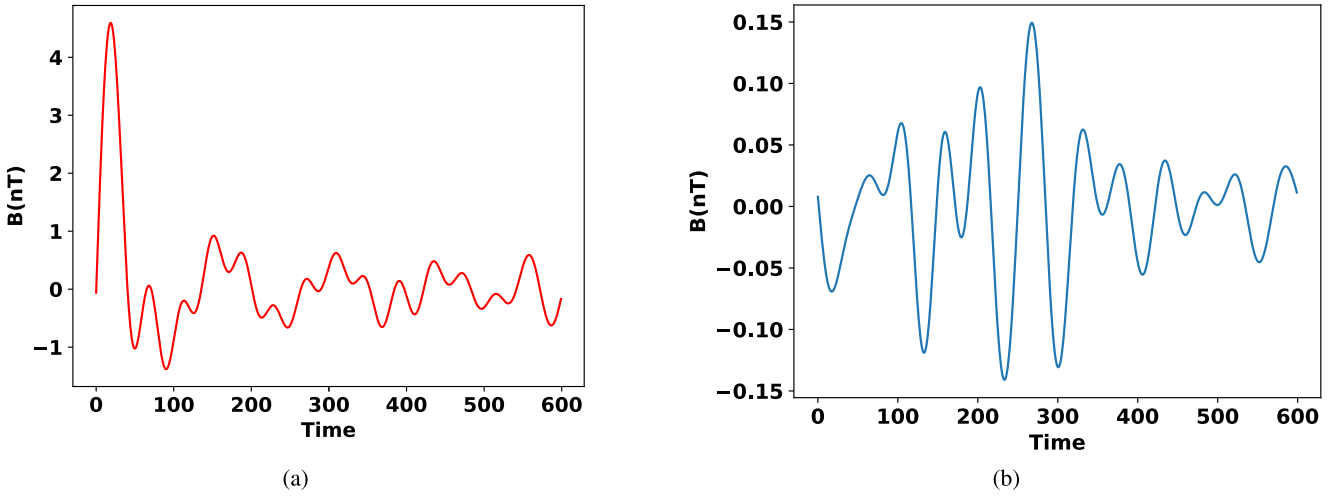

FIGURE 2. Non-Pi2 sample (a) and Pi2 sample (b).

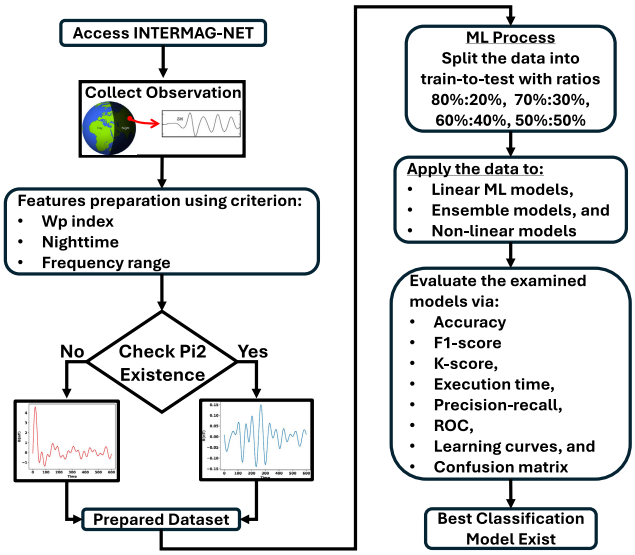

FIGURE 3. Employed system model from data collection to the best model of classification.

opposite class is defined by 0. The likelihood of this approach is $Pr(y = 1|\mathbf{x})$ that expresses the output $y = \{0, 1\}$, given the input features $\mathbf{x}$:

$$Pr(y = 1|\mathbf{x}) = \frac{1}{1 + e^{-(w_1x_1+w_2x_2+\ldots+w_mx_m+b)}}, \quad (1)$$

$$Pr(y = 0|\mathbf{x}) = 1 - Pr(y = 1|\mathbf{x}). \quad (2)$$

2) SUPPORT VECTOR MACHINE

SVM are supervised learning algorithms primarily used for discrimination. It is well-known that SVM relies on searching for the best hyperplane separating the data points of different labels in a high-dimensional space [42]. In an n-dimensional space, a hyperplane can be defined by:

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \quad (3)$$

Algorithm 1 Proposed Algorithm for Pi2 Classification Using ML.

Input Read all buses' datasets **while** *Best model exist via the evaluation metrics* **do**
  Pi2 Classification;
  Access to the magnetometer network INTERMAG-NET
  Collect the magnetometers' observations;
  Focus on local nighttime (22:00 to 02:00);
  Evaluate the W_p index as in Table 1;
  Filter the data based frequency range (6.7 to 25 $\times$ 10^{-3} Hz);
  Detecting signal peaks within the first 150 s and min. amplitude of 1 nT
  if *Detection is successful* **then**
    Signal labeled as Pi2
  else
    Labeled as Non-Pi2
  end
  List the labels based on the corresponding event type (Pi2 and Non-Pi2);
  Use five different train-test-split ratios (80%:20%, 70%:30%, 60%:40%, 50%:50%);
  Apply the fourteen ML models to the prepared datasets;
  Hyper-parameters tuning;
  Evaluate the models via seven metrics discussed in Sections IV-A, IV-C, IV-D, IV-B, IV-E,IV-F, and IV-G;
end
Output: Best performance of Pi2 Classification Exist.

where $\mathbf{w}$ denotes the weight vector (normal to the hyperplane), $\mathbf{x}$ expresses the input feature vector, and b represents the bias term. Accordingly, the SVM target is to maximize the distance between the hyperplane, the margin, and the closest data points from either label (support vectors). The

optimization problem can be formulated as:

$$\begin{aligned} &\text{maximize} \quad \frac{2}{\|\mathbf{w}\|}, \\ &\text{s.t} \\ &y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1, \quad \forall i, \end{aligned} \tag{4}$$

where y_i is the true label for the i-th data point (either $+1$ or -1), and $\mathbf{x}_i$ is the feature vector for the i-th data point. In practice, the data may not be perfectly separable. To handle this, SVM introduces a soft margin, allowing some misclassifications. The optimization problem then becomes:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i, \\ &\text{s.t} \\ &y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \forall i, \end{aligned} \tag{5}$$

where ξ_i are slack variables representing the degree of misclassification, and C is a regularization parameter that controls the margin maximization and the classification error minimization challenge. SVMs can also efficiently handle non-linear decision boundaries using the kernel trick, which transforms the input space into a higher-dimensional space.

3) LINEAR DISCRIMINATION ANALYSIS

Linear Discriminant Analysis (LDA) is a classification algorithm based on finding a linear combination of features that best separates two or more classes. It assumes that data from each class is normally distributed with a shared covariance matrix [43]. LDA is commonly employed for reducing dimension and task discrimination. The discriminant function for class C_k is:

$$\delta_k(x) = x^\top \Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^\top \Sigma^{-1}\mu_k + \ln(P(C_k)), \tag{6}$$

where x is the input feature vector, μ_k represents the mean vector of class C_k, denotes the shared covariance matrix across all classes, and $P(C_k)$ expresses the prior probability of class C_k. The predicted class $\hat{y}$ is:

$$\hat{y} = \arg\max_k \delta_k(x). \tag{7}$$

4) RIDGE

The Ridge classifier is a variant of Ridge Regression adapted for classification tasks. It works by minimizing a regularized least squares objective, which helps prevent overfitting by penalizing large coefficients. This classifier is especially useful when the features are highly correlated or the features exceed the number of samples [44]. Ridge Classifier applies a linear decision rule, solving the following optimization problem:

$$\hat{\beta} = \arg\min_\beta \left\{ \sum_{i=1}^{N} \left(y_i - \mathbf{x}_i^\top \beta\right)^2 + \alpha\|\beta\|_2^2 \right\}, \tag{8}$$

where y_i is encoded as -1 or 1 for binary classification, $\mathbf{x}_i$ denotes the feature vector for sample i, β expresses the coefficient vector, α presents the regularization strength, controlling the penalty on large coefficients, and $\|\beta\|_2^2$ is the L2 norm (sum of squared coefficients). The decision boundary is determined by:

$$\hat{y} = \text{sign}\left(\mathbf{x}^\top \hat{\beta}\right). \tag{9}$$

B. ENSEMBLE ML MODES

1) K-NEAREST NEIGHBORS

The KNN model is an instance-based algorithm that retains the entire training dataset and does not require an individual training stage. During prediction, it uses a distance metric (e.g., Euclidean distance) to identify the K-nearest neighbors of a new data point, which then votes on its category [45]. This makes KNN well-suited for dynamic datasets and scenarios with nonlinear decision boundaries, as it can observe local paradigms and adapt the data changes. This model's advantage is its simplicity, implementation, and capability to adapt span of data types, including numerical, categorical, and mixed formats. It is commonly used in applications, e.g., text and image recognition, anomaly detection, and recommendation systems. However, as the training dataset grows, KNN's computational demands increase since it requires calculating the spacing between each new data point, which can become inefficient with larger datasets as expressed by:

$$\mathcal{D} = \{(\mathbf{v}_1, c_1), (\mathbf{v}_2, c_2), \ldots, (\mathbf{v}_m, c_m)\}, \tag{10}$$

where $\mathbf{v}_j$ denotes a feature vector and c_j signifies the associated class label.

2) QUADRATIC DISCRIMINATION ANALYSIS

QDA enhances the principles of linear discriminant analysis (LD) by modeling each class's probability distribution with a quadratic function [46], [47], [48]. Unlike LD, which assumes a shared covariance matrix, QDA allows each class to have its own covariance structure, making it suitable for datasets with non-linear decision boundaries and complex feature relationships. This flexibility enables QDA to accurately represent scenarios where classes exhibit diverse covariance patterns, such as in medical diagnoses, where the covariance structure of diseases may differ due to various biomarkers. By accommodating these variations, QD offers more precise classification results compared to LD.

3) GAUSSIAN NAIVE BAYES

The Gaussian NB classifier is a probabilistic model based on Bayes' theorem and assumes that the features follow a Gaussian (normal) distribution. It is called ''naive'' because it assumes conditional independence between features given the class label [49]. The Gaussian NB advantages involve simplicity and computational efficiency. On the other hand, the assumption of feature independence adopted in this model can reduce accuracy when features are strongly correlated. Besides, it is sensitive to the Gaussian distribution assumption

for feature values. The posterior probability for a class C_k given a data point $\mathbf{x}$ is computed as:

$$P(C_k \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid C_k)P(C_k)}{P(\mathbf{x})}, \tag{11}$$

where $P(C_k \mid \mathbf{x})$ is posterior probability of class C_k given $\mathbf{x}$, $P(\mathbf{x} \mid C_k)$ represents Likelihood of $\mathbf{x}$ for class C_k, modeled as a Gaussian distribution:

$$P(x_i \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x_i - \mu_k)^2}{2\sigma_k^2}\right), \tag{12}$$

where μ_k is the mean of the feature for class C_k, σ_k^2 denotes the variance of the feature for class C_k, $P(C_k)$ presents the prior probability of class C_k, and $P(\mathbf{x})$ expresses evidence, which is constant across all classes and often omitted in classification.

4) DECISION TREE

There are nodes and branches in a decision tree (DT). Every leaf node indicates a class label or an expected value, every branch indicates the decision's result, and every internal node reflects a choice based on a feature.

Let $\mathcal{T} = \{(\mathbf{z}_1, \ell_1), (\mathbf{z}_2, \ell_2), \ldots, (\mathbf{z}_m, \ell_m)\}$ represent the training set, where $\mathbf{z}_j \in \mathbb{R}^p$ is the feature vector, and ℓ_j is the corresponding target label. The DT is built by recursively splitting data at each node using the feature and threshold that optimize the chosen criterion (Gini, entropy, or variance reduction) [50]. Until a halting condition is satisfied, such as the maximum depth, the minimum number of node samples, or the satisfactory classification.

C. NON-LINEAR ML MODES

To effectively classify Pi2 and non-Pi2 magnetic signals, we selected five well-established non-linear machine learning models: Random Forest, Extra Trees (ET), Gradient Boosting (GB), AdaBoost (Ada), and extreme gradient boost (XGB). These classifiers are known for their ability to handle complex, non-linear patterns and noisy data, which are characteristic of geomagnetic time series. Their robustness, adaptability, and strong performance in signal classification tasks make them suitable choices for detecting the subtle distinctions between Pi2 events and background fluctuations.

1) RANDOM FOREST

This classifier is an ensemble learning method integrating several DTs to enhance discrimination accuracy and control overfitting. It works by constructing several DTs through training and testing the label that is the mode of the labels estimated by the individual trees [51]. The advantages of RF are its robustness to overfitting due to the aggregation of multiple trees, its handling of both classification and regression tasks effectively, and its capability to handle large datasets and high-dimensional feature spaces. Conversely, it requires more computational resources compared to simpler models, and it is less interpretable than a single DT. It is worth mentioning that each DT in the forest is trained on a random subset of the dataset using a technique called *bagging* (Bootstrap Aggregating), where a random subset of features is selected at each split in the tree, and predictions are aggregated (majority vote for classification or averaging for regression). For a classification task, the predicted label $\hat{y_p}$ for an input $\mathbf{x}$ is given by:

$$\hat{y_p} = \text{mode}\left\{PF_1(\mathbf{x}), PF_2(\mathbf{x}), \ldots, PF_T(\mathbf{x})\right\}, \tag{13}$$

where $PF_i(\mathbf{x})$ is the i-th DT prediction, and T denotes sum of DTs in the forest.

2) EXTRA TREES

The ET classifier is an ensemble learning method that builds multiple DTs and aggregates their predictions to improve accuracy and robustness. It differs from Random Forests by introducing additional randomization in the tree-building process to further reduce variance [52]. The advantages of DT can be summarized as reducing variance compared to a single DT, being faster to train than Random Forest due to randomized splits, and handling large datasets and high-dimensional feature spaces effectively. On the other hand, DT can still suffer from overfitting if the dataset is noisy, and less interpretable than a single DT, which is given by Eq. 13.

3) GRADIENT BOOSTING

This classifier builds models sequentially as each corrects the errors of its predecessor. It uses DTs as weak learners and combines them to create a strong predictive model. The process minimizes a loss function using gradient descent [53]. The benefits of GB are as follows: Highly accurate and widely used in machine learning competitions, handling various types of loss functions and supporting custom ones, and effective for both regression and classification tasks. Conversely, GB has an intensive computation compared to simpler models, more sensitivity to hyperparameters, especially the number of trees and learning rate, and is prone to overfitting if the number of trees is too large. For a given dataset (x_i, y_i), the predicted value $\hat{F}_M(x)$ after M iterations (trees) is:

$$\hat{F}_M(x) = \sum_{m=1}^{M} b \cdot PF_m(x), \tag{14}$$

where $PF_m(x)$ is the prediction form of the m-th weak learner (DT), b denotes the learning rate, which controls the contribution of each tree, and M represents the total number of trees in the model. The m-th tree is trained to minimize the loss function $L(y, F(x))$ using gradient descent:

$$PF_m(x) = \arg\min_{PF} \sum_{i=1}^{M} \left(-\frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)} \cdot PF(x_i)\right). \tag{15}$$

4) ADABOOST

The Ada classifier integrates multiple weak classifiers, such as DTs, to produce a robust classifier. It assigns weights to instances, emphasizing those misclassified by previous classifiers, and updates these weights iteratively to improve accuracy [54]. The Ada advantages are its effectiveness with weak classifiers in improving their accuracy significantly, its robustness to overfitting if properly tuned, and its simplicity to implement is widely used. The disadvantage is its sensitivity to noisy data and outliers due to its emphasis on difficult examples. The prediction of Ada is given by:

$$\hat{y} = \text{sign}\left(\sum_{m=1}^{M} \alpha_m PF_m(x)\right), \tag{16}$$

where α_m presents the weight assigned to the m-th classifier, calculated as:

$$\alpha_m = \frac{1}{2} \ln\left(\frac{1 - ce_m}{ce_m}\right), \tag{17}$$

where ce_m is the classification error of the m-th classifier, and M is the total number of weak classifiers. The instance weights w_i are updated after each iteration:

$$w_i^{(m+1)} = w_i^{(m)} \exp\left(\alpha_m \cdot \mathbb{I}[y_i \neq h_m(x_i)]\right), \tag{18}$$

where $\mathbb{I}[\cdot]$ represents the indicator function, which is only "1" if the condition is satisfied.

5) EXTREME GRADIENT BOOSTING

XGB is particularly valued for its speed, accuracy, and versatility in handling both structured and unstructured data types [55]. XGB falls within the category of gradient boosting algorithms, which are ensemble methods that sequentially construct models. Specifically, in gradient boosting, each new model is designed to correct the errors made by its predecessors. The final prediction is obtained by aggregating the outputs of all individual models. The general prediction formula is expressed as follows:

$$\hat{y}_i^{(t)} = \Gamma^{(t)}(r_i) = \Gamma^{(t-1)}(r_i) + PF_t(r_i), \tag{19}$$

where $\Gamma^{(t)}(r_i)$ presents the prediction from the ensemble after t iterations, and $PF_t(r_i)$ is the prediction from the t-th model, typically a weak learner like a DT.

6) LIGHT GRADIENT BOOSTING

The LGB is a framework designed for efficiency and scalability with large datasets. LGB differs from traditional gradient boosting implementations by using a leaf-wise growth strategy and histogram-based learning, making it faster and more memory-efficient [56]. The predicted value is calculated by Eq. 14, and each tree is trained to minimize a loss function $L(y, F(x))$ using gradient descent as calculated by Eq. 15.

IV. EVALUATION METHODOLOGY

A. ACCURACY

A simple and obvious metric is accuracy [57]. It provides you with the accuracy percentage of the predictions made by the entire model, which is determined by:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{20}$$

where FP is the false positive instances that are incorrectly predicted as positive, TN denotes the true negative ones, indicating the negative instances, which are predicted correctly as negative, and finally, FN represents the false negative ones, which indicate the positive instances incorrectly predicted as negative.

It is crucial to remember, though, that in situations when there is an imbalance in the courses, accuracy may be deceptive. A model that consistently predicts the dominant class in a dataset, for instance, may have high accuracy but may struggle to identify the minority class. Additional metrics, such as recall, precision, or the F1-score, might offer a more impartial assessment of the algorithm's effectiveness under certain circumstances.

B. PRECISION AND RECALL

1) METRIC OF PRECISION

A factor called precision is used to assess how well the model predicts favorable results. It is the proportion of positively predicted cases that were accurately forecasted of all the positively predicted cases. Furthermore, precision shows the proportion of positively anticipated events. A high degree of accuracy implies a low number of false positive errors [58]. It is crucial in scenarios where the expense of FPs is substantial, like spam detection or medical diagnosis. It is provided by:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{21}$$

2) METRIC OF RECALL

Recall measures the way each successful example is captured by the model. When a model has a high recall, it means it creates fewer false negatives, which is important when it comes to situations where missing a positive instance might be expensive, like in illness screening or fraud detection. Recall, often referred to as sensitivity or true positive rate, quantifies the model's capacity to detect every positive case [58] accurately. It is given by:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{22}$$

C. F1 SCORE

When memory and precision are equally crucial, the F1 Score offers a fair assessment. It is quite helpful when there is an unequal distribution of classes or when there are substantial expenses associated with both FPs and FNs. A high F1 Score indicates that the model accurately predicts the positive outcomes while successfully identifying all

positive examples, demonstrating a solid balance between precision and recall [59]. It resolves the trade-off between precision and recall, which is provided by:

$$\text{F1 Score} = \frac{TP}{TP + 0.5 \times (FP + FN)}. \tag{23}$$

D. COHEN'S KAPPA SCORE

A statistical metric called Cohen's Kappa Score evaluates the degree of conformity between two classifiers over what would be predicted by chance. It is frequently used to determine how well classification models work, particularly when there is an imbalance between the classes. When working with unbalanced datasets, Cohen's Kappa Score offers a more reliable indicator of classifier performance than accuracy alone. It considers the likelihood that the model will make accurate predictions by chance, whereas accuracy only counts the percentage of correct predictions [60]. Perfect agreement is indicated by a Kappa score of 1, while agreement is no better than chance when the score is 0. Inverse values signify a lower degree of agreement than expected, as given by:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{24}$$

where p_o (Observed Agreement) is the proportion of times the raters (or the model and ground truth) agree and can be represented by Eq. 20, p_e (Expected Agreement) is the proportion of times the evaluators can be expected to agree by chance, as given by:

$$p_e = \frac{(TP + FP) \times (TP + FN)}{(TP + TN + FP + FN)^2} + \frac{(FN + TN) \times (FP + TN)}{(TP + TN + FP + FN)^2}. \tag{25}$$

E. LEARNING CURVE

It is trained on progressively larger datasets, with training and validation metrics (such as accuracy or loss) plotted against the size of the training data. The horizontal axis represents the quantity of training data, while the vertical axis displays the performance metric. Learning curves are useful for diagnosing model behavior: they can reveal underfitting (characterized by high errors in both training and validation), overfitting (where the training error is low but the validation error is high), or optimal performance (indicated by low and similar errors in both training and validation) [61]. The curve is defined as:

$$P = N \cdot K^{-\beta}, \tag{26}$$

where N represents the cumulative number of units produced up to the current point, P is the cumulative average production time per unit, and β is the learning rate constant.

F. RECEIVER OPERATING CHARACTERISTIC (ROC)

It illustrates the trade-off between the TP rate (TPR or sensitivity) and the FP rate (FPR) or (1 - specificity) across different threshold values. TPR measures the proportion of actual positives correctly identified. Besides, the FPR is used to quantify the ROC, which is the proportion of actual negatives that are incorrectly identified. FPR and TPR can be derived by Eqs. 21 and 22, respectively.

The ROC curve plots TPR against FPR at various threshold settings. An ideal classifier would have a TPR of 1 and an FPR of 0, resulting in a curve that hugs the top left corner of the plot. Then, the AUC quantifies the overall performance of the classifier. An AUC of 0.5 indicates no discrimination (random guessing), while an AUC of 1.0 indicates perfect discrimination:

$$\text{AUC} = \int_0^1 \text{TPR}\, d(\text{FPR}). \tag{27}$$

G. CONFUSION MATRIX

A confusion matrix is a tool used to assess the performance of a classification ML model by comparing its predictions with the actual outcomes. It is structured into four key components: TP, TN, FP, and FN. This matrix provides a comprehensive overview of the model's classification accuracy, enabling the calculation of various performance metrics such as F1-score, accuracy, recall, and precision. The confusion matrix is especially useful for identifying the types of errors the model makes, making it an essential tool in tasks of binary and multi-class classification [62].

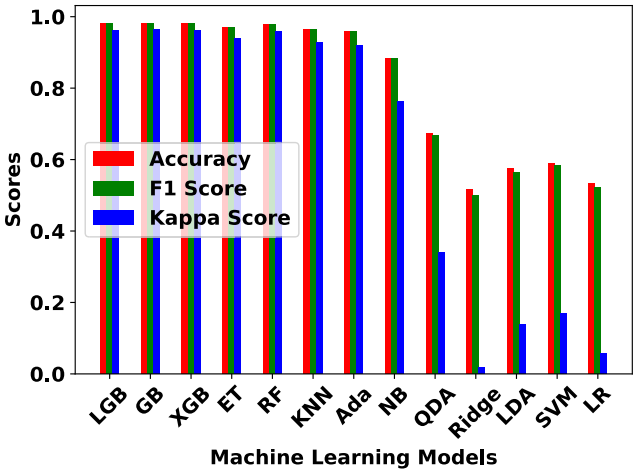

FIGURE 4. Accuracy measurements using overall accuracy, F1 score, and kappa score.

V. RESULTS AND DISCUSSION

This paper outlines three key steps in the analytical approach to classifying Pi2 observations: selecting relevant features, creating a statistical model using these features, and examining the model against the primary indicators of the dominant classes. In this regard, accurate classification contributes to efficient observability. With the Scikit-Learn module, Python is used to implement various classification techniques. It is worth mentioning that we have tested the classifiers using a

TABLE 2. Optimized model parameters.

Model	Parameters
GB	*ccp_alpha* = 0.0, *criterion* = *friedman_mse*, *init* = *None*, *learning_rate* = 0.1, *loss* = *log_loss*, *max_depth* = 3, *max_features* = *None*, *max_leaf_nodes* = *None*, *min_impurity_decrease* = 0.0, *min_samples_leaf* = 1, *min_samples_split* = 2, *min_weight_fraction_leaf* = 0.0, *n_estimators* = 100, *n_iter_no_change* = *None*, *random_state* = 123, *subsample* = 1.0, *tol* = 0.0001, *validation_fraction* = 0.1, *verbose* = 0, *warm_start* = *False*

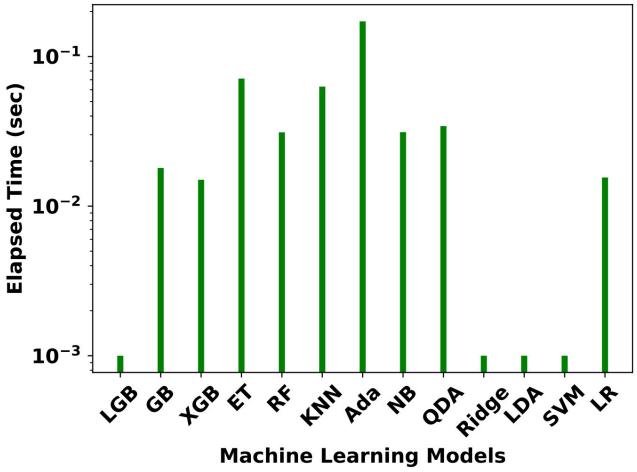

FIGURE 5. Comparison of the employed models' execution time.

10-fold cross-validation function. In addition, the optimized hyperparameter configuration of the best model is listed in Table 2.

To ensure the model's efficacy, we have adopted four different train-test-split ratios (80%:20%, 70%:30%, 60%:40%, 50%:50%). The trained model was applied to the testing set to evaluate whether the validation set effectively represents the testing set. It is worth mentioning that the proposed approach proved beneficial regardless of the train-test split ratio. Accordingly, hereafter, we present the results of the split-ratio of 80% to 20% between the training and testing datasets, respectively.

Figure 4 shows the obtained model's accuracy of Pi2 classification. More particularly, in this stage, the model aims to discriminate between the Pi2 and Non-Pi2. Besides, we have examined the impact of hyperparameter tuning on model performance, which demonstrated the models' effectiveness. It is clearly shown that the excellent performance of the GB among the other benchmark models in both overall accuracy, F1-score, and kappa score, with 98.21%, 98.21%, and 96.4%, respectively. Meanwhile, the other benchmarks achieved a lower performance as compared to the GB with the three evaluation metrics. Besides, Figure 5 shows a comparison between the elapsed time of all employed models (linear, ensemble, and non-linear). It is clearly shown that the GB

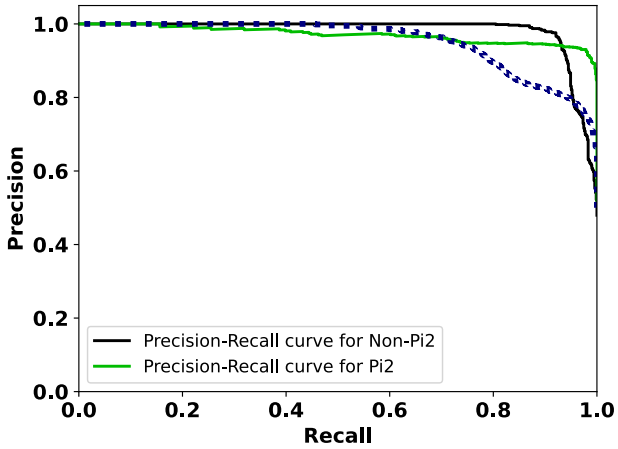

FIGURE 6. Precision recall curve of testset via the GB model.

is tested in a quite short time. Although some other models achieved little elapsed time compared to the GB, the GB is considered the best-performing model due to its accuracy and sufficient execution time. Hereafter, we present the rest of the evaluation metrics of the Pi2 classification performance based on the best model (GB).

In Figure 6, precision-recall has demonstrated that the GB model presents a superior classification capability with 100% of all data buses. Moreover, the learning curve highlights the effectiveness of the model and reveals whether overfitting exists or not. Accordingly, we have examined the proposed model with subsets of the utilized data to ensure that "no overfitting" exists achieveing the highest classification accuracy regardless of the subset utilized as shown in Figure 7. It is clearly shown that the validation curve stays close to 100%, reflecting excellent discrimination accuracy. The analogy of the training curve and validation curve suggests that the validation accuracy could continue to improve with more data.

Similarly, the ROC curve confirms the outstanding performance obtained by the GB as depicted in Figure 8. To present the results more effectively and demonstrate the robustness of the proposed model, a confusion matrix was employed to visually display the classification performance of the predicted classes, as shown in Figure 9. Mostly, the obtained accuracy of data bus discrimination reached an accuracy of 98.21% from all buses, reflecting the models' robustness for Pi2 pulsation detection.

VI. CONCLUSION

This study was conducted to propose a method to detect Pi2 magnetic pulsation using an automated ML approach as an alternative to the discontinued index. In this study, the horizontal component of geomagnetic field data obtained from the INTERMAGNET network was utilized. Through the ML framework, the training of various algorithms and hyperparameter tuning was carried out automatically. To ensure a robust comparison, multiple linear, ensemble, and

(a) No subset (b) Using 70% of the Dataset (c) Using 50% of the Dataset

FIGURE 7. Learning curve of testset via the GB model considering subsets examination.

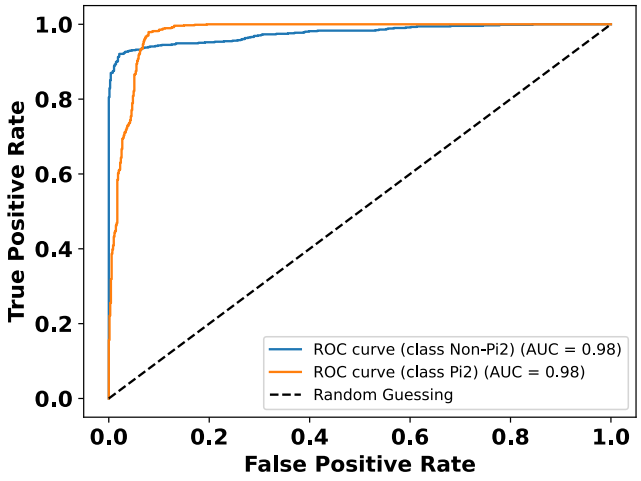

FIGURE 8. ROC Curve of testset via the GB model.

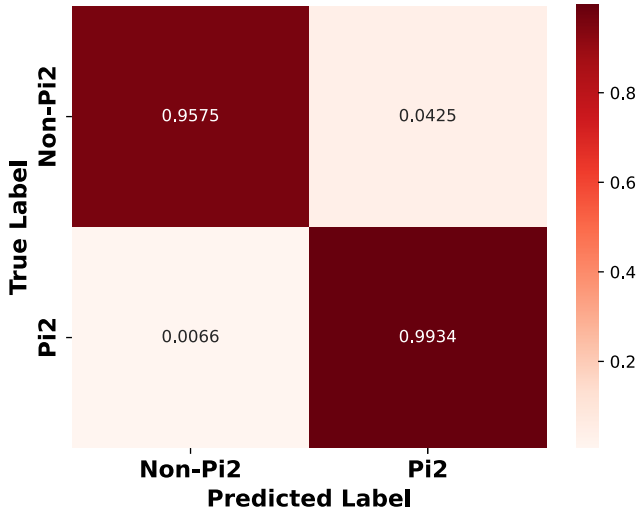

FIGURE 9. Confusion matrix of testset via the GB model.

non-linear classification models were examined. The system relies on continuous real-time data streaming from INTERMAGNET. However, this system is limited by the detection time gap, which can reach up to 6 hours. This limitation arises from the time gap between when the data is observed and when it is published by INTERMAGNET. By shortening this time gap, the system could be improved to achieve true real-time detection. To ensure the proposed model's effectiveness, we have evaluated it using seven metrics: overall accuracy, F1-score, Cohen's kappa score, elapsed time, precision recall, ROC, learning curve, and confusion matrix. The employed GB approach has outperformed the benchmarks with an accuracy of 98.21%. The outstanding performance of the GB model, particularly in terms of accuracy, highlights its potential as a reliable tool for enhancing the predictability of the Pi2. The limitation of the proposed approach involves considering only the nighttime data from different stations distributed around the globe at different longitudes. In future work, we aim to examine our model using different data qualities and studying the impact of data latency.

ACKNOWLEDGMENT

The work in this paper was supported, in part, by the open access program and project FRG24-C-E66 from the American University of Sharjah. This paper represents the opinions of the author(s) and does not mean to represent the position or opinions of the American University of Sharjah. Acknowledgements are also extended to the members of the 553 International Space Science Institute (ISSI) Bern project titled "CSES and Swarm Investigation of the Generation Mechanisms of Low Latitude Pi2 Waves," led by Essam Ghamry and Zeren Zhima, for their discussions and valuable insights.

REFERENCES

[1] D. N. Baker, T. I. Pulkkinen, V. Angelopoulos, W. Baumjohann, and R. L. McPherron, "Neutral line model of substorms: Past results and present view," *J. Geophys. Res., Space Phys.*, vol. 101, no. A6, pp. 12975–13010, Jun. 1996.

[2] R. L. McPherron, "Magnetospheric substorms," *Rev. Geophysics*, vol. 17, no. 4, pp. 657–681, Jun. 1979.

[3] M. Maimaiti, B. Kunduri, J. M. Ruohoniemi, J. B. H. Baker, and L. L. House, "A deep learning-based approach to forecast the onset of magnetic substorms," *Space Weather*, vol. 17, no. 11, pp. 1534–1552, Nov. 2019.

[4] G. Anagnostopoulos, A. Karkanis, A. Kampatagis, P. Marhavilas, S.-A. Menesidou, D. Efthymiadis, S. Keskinis, D. Ouzounov, N. Hatzigeorgiu, and M. Danikas, "Ground electric field, atmospheric weather and electric grid variations in northeast Greece influenced by the March 2012 solar activity and the moderate to intense geomagnetic storms," *Remote Sens.*, vol. 16, no. 6, p. 998, Mar. 2024.

[5] J. A. Jacobs, Y. Kato, S. Matsushita, and V. A. Troitskaya, "Classification of geomagnetic micropulsations," *Geophys. J. Roy. Astronomical Soc.*, vol. 8, no. 3, pp. 341–342, Apr. 2007.

[6] E. Ghamry, D. Marchetti, A. Yoshikawa, T. Uozumi, A. De Santis, L. Perrone, X. Shen, and A. Fathy, "The first Pi2 pulsation observed by China seismo-electromagnetic satellite," *Remote Sens.*, vol. 12, no. 14, p. 2300, Jul. 2020.
[7] T. Saito, T. Sakurai, and Y. Koyama, "Mechanism of association between Pi2 pulsation and magnetospheric substorm," *J. Atmos. Terr. Phys.*, vol. 38, no. 12, pp. 1265–1277, Dec. 1976.
[8] T. Saito, K. Yumoto, and Y. Koyama, "Magnetic pulsation Pi2 as a sensitive indicator of magnetospheric substorm," *Planet. Space Sci.*, vol. 24, no. 11, pp. 1025–1029, Nov. 1976.
[9] J. V. Olson, "Pi2 pulsations and substorm onsets: A review," *J. Geophys. Res., Space Phys.*, vol. 104, no. A8, pp. 17499–17520, Aug. 1999.
[10] A. Keiling and K. Takahashi, "Review of Pi2 models," *Space Sci. Rev.*, vol. 161, nos. 1–4, pp. 63–148, Nov. 2011.
[11] E. Ghamry, K. Kim, H. Kwon, D. Lee, J. Park, J. Choi, K. Do, W. S. Kurth, C. Kletzing, J. R. Wygant, and J. Huang, "Simultaneous Pi2 observations by the van Allen probes inside and outside the plasmasphere," *J. Geophys. Res. Space Phys.*, vol. 120, no. 6, pp. 4567–4575, May 2015.
[12] P. R. Sutcliffe, "Pi2 band activity at low latitudes during non-substorm intervals," *Geophys. Res. Lett.*, vol. 37, no. 5, pp. 1–5, Mar. 2010.
[13] E. Ghamry, "Pi2 pulsations during extremely quiet geomagnetic condition: Van Allen probe observations," *J. Astron. Space Sci.*, vol. 34, no. 2, pp. 111–118, Jun. 2017.
[14] M. Nosé, T. Iyemori, L. Wang, A. Hitchman, J. Matzka, M. Feller, S. Egdorf, S. Gilder, N. Kumasaka, K. Koga, H. Matsumoto, H. Koshiishi, G. Cifuentes-Nava, J. J. Curto, A. Segarra, and C. Çelik, "Wp index: A new substorm index derived from high-resolution geomagnetic field data at low latitude," *Space Weather*, vol. 10, no. 8, pp. 1–12, Aug. 2012.
[15] M. Nosé, T. Iyemori, M. Takeda, T. Kamei, D. K. Milling, D. Orr, H. J. Singer, E. W. Worthington, and N. Sumitomo, "Automated detection of pi 2 pulsations using wavelet analysis: 1. Method and an application for substorm monitoring," *Earth, Planets Space*, vol. 50, no. 9, pp. 773–783, Jun. 2014.
[16] M. Nosé, "Automated detection of pi 2 pulsations using wavelet analysis: 2. An application for dayside pi 2 pulsation study," *Earth, Planets Space*, vol. 51, no. 1, pp. 23–32, Jun. 2014.
[17] W. Mo, J. Wang, G. Yuan, D. Cao, and G. Bai, "Application of machine learning in magnetocaloric materials: A review," *Mater. Today Commun.*, vol. 44, Mar. 2025, Art. no. 111933.
[18] S. Pantopoulou, M. Weathered, D. Lisowski, L. H. Tsoukalas, and A. Heifetz, "Temporal forecasting of distributed temperature sensing in a thermal hydraulic system with machine learning and statistical models," *IEEE Access*, vol. 13, pp. 10252–10264, 2025.
[19] M. S. Abdalzaher, M. Sami Soliman, and S. M. El-Hady, "Seismic intensity estimation for Earthquake early warning using optimized machine learning model," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5914211.
[20] M. S. Abdalzaher, S. S. R. Moustafa, H. E. A. Hafiez, and W. F. Ahmed, "An optimized learning model augment analyst decisions for seismic source discrimination," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022.
[21] R. Shao, C. Yang, Q. Li, L. Xu, X. Yang, X. Li, M. Li, Q. Zhu, Y. Zhang, Y. Li, Y. Liu, Y. Tang, D. Liu, S. Yang, and H. Li, "AllSpark: A multimodal spatiotemporal general intelligence model with ten modalities via language as a reference framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025.
[22] M. S. Abdalzaher, M. Krichen, S. S. R. Moustafa, and M. Alswailim, "Using machine learning for earthquakes and quarry blasts discrimination," in *Proc. 20th ACS/IEEE Int. Conf. Comput. Syst. Appl. (AICCSA)*, Dec. 2023, pp. 1–6.
[23] M. S. Abdalzaher, S. S. R. Moustafa, W. Farid, and M. M. Salim, "Enhancing analyst decisions for seismic source discrimination with an optimized learning model," *Geoenvironmental Disasters*, vol. 11, no. 1, p. 23, Aug. 2024.
[24] E. L. Habbak, M. S. Abdalzaher, A. S. Othman, and H. Mansour, "Enhancing the classification of seismic events with supervised machine learning and feature importance," *Sci. Rep.*, vol. 14, no. 1, p. 30638, Dec. 2024.
[25] S. Cheng, H. Zhang, and T. Alkhalifah, "Self-supervised seismic resolution enhancement," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, 2025.
[26] M. S. Abdalzaher, M. Sami Soliman, and M. M. Fouda, "Using deep learning for rapid earthquake parameter estimation in single-station single-component earthquake early warning system," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024.
[27] Q. Zhe, W. Gao, C. Zhang, G. Du, Y. Li, and D. Chen, "A hyperspectral classification method based on deep learning and dimension reduction for ground environmental monitoring," *IEEE Access*, vol. 13, pp. 29969–29982, 2025.
[28] S. Rizwan, C. Ken Nee, and S. Garfan, "Identifying the factors affecting Student academic performance and engagement prediction in MOOC using deep learning: A systematic literature review," *IEEE Access*, vol. 13, pp. 18952–18982, 2025.
[29] M. S. Abdalzaher, M. S. Soliman, S. M. El-Hady, A. Benslimane, and M. Elwekeil, "A deep learning model for Earthquake parameters observation in IoT system-based earthquake early warning," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8412–8424, Jun. 2022.
[30] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated Machine Learning: Methods, Systems, Challenges*. Cham, Switzerland: Springer, 2019.
[31] Y. Watabe and T. Shibuya, "Nonlinear value function approximation method with easy hyperparameter tuning and convergence guarantee," *IEEE Access*, vol. 13, pp. 30117–30126, 2025.
[32] D.-P. Tran, G.-N. Nguyen, and V.-D. Hoang, "Hyperparameter optimization for improving recognition efficiency of an adaptive learning system," *IEEE Access*, vol. 8, pp. 160569–160580, 2020.
[33] X. He, K. Zhao, and X. Chu, "AutoML: A survey of the state-of-the-art," *Knowledge-Based Syst.*, vol. 212, Jan. 2021, Art. no. 106622.
[34] M. Crack, C. J. Rodger, M. A. Clilverd, D. H. Mac Manus, I. Martin, M. Dalzell, S. P. Subritzky, N. R. Watson, and T. Petersen, "Even-Order harmonic distortion observations during multiple geomagnetic disturbances: Investigation from new Zealand," *Space Weather*, vol. 22, no. 5, pp. 1–19, May 2024.
[35] N. Ganushkina, B. M. Swiger, S. Dubyagin, J.-C. Matéo-Vélez, M. W. Liemohn, A. Sicard, and D. Payan, "Worst-Case severe environments for surface charging observed at LANL satellites as dependent on solar wind and geomagnetic conditions," *Space Weather*, vol. 19, no. 9, pp. 1–27, Aug. 2021.
[36] (2024). *Index Data*. Accessed: Jun. 1, 2024. [Online]. Available: https://www.isee.nagoya-u.ac.jp/nose.masahito/s-cubed
[37] (2024). *Intermagnet Data*. Accessed: Jun. 1, 2024. [Online]. Available: https://intermagnet.org/
[38] N. Thomas, G. Vichare, A. K. Sinha, and R. Rawat, "Low-latitude Pi2 oscillations observed by polar low Earth orbiting satellite," *J. Geophys. Res., Space Phys.*, vol. 120, no. 9, pp. 7838–7856, Sep. 2015.
[39] M. Teramoto, Y. Miyoshi, A. Matsuoka, Y. Kasahara, A. Kumamoto, F. Tsuchiya, M. Nosé, S. Imajo, M. Shoji, S. Nakamura, M. Kitahara, and I. Shnohara, "Off-equatorial Pi2 pulsations inside and outside the plasmapause observed by the Arase satellite," *J. Geophys. Res., Space Phys.*, vol. 127, no. 1, p. 2021, Jan. 2022.
[40] X. Zou, Y. Hu, Z. Tian, and K. Shen, "Logistic regression model optimization and case analysis," in *Proc. IEEE 7th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT)*, Oct. 2019, pp. 135–139.
[41] Y. Yang and M. Loog, "A benchmark and comparison of active learning for logistic regression," *Pattern Recognit.*, vol. 83, pp. 401–415, Nov. 2018.
[42] Y.-W. Chang and C.-J. Lin, "Feature ranking using linear SVM," in *Causation and Prediction Challenge*. Hong Kong, China: PMLR, 2008, pp. 53–64.
[43] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction To Statistical Learning*, vol. 112, no. 1. New York, NY, USA: Springer, 2013.
[44] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Trans. Med. Imag.*, vol. 23, no. 4, pp. 501–509, Apr. 2004.
[45] S. TAN, "An effective refinement strategy for KNN text classifier," *Expert Syst. Appl.*, vol. 30, no. 2, pp. 290–298, Feb. 2006.
[46] A. Tharwat, "Linear vs. Quadratic discriminant analysis classifier: A tutorial," *Int. J. Appl. Pattern Recognit.*, vol. 3, no. 2, p. 145, 2016.
[47] S. Bose, A. Pal, R. SahaRay, and J. Nayak, "Generalized quadratic discriminant analysis," *Pattern Recognit.*, vol. 48, no. 8, pp. 2676–2684, Aug. 2015.
[48] N.-D. Hoang and D. T. Bui, "Predicting Earthquake-induced soil liquefaction based on a hybridization of kernel Fisher discriminant analysis and a least squares support vector machine: A multi-dataset study," *Bull. Eng. Geol. Environ.*, vol. 77, no. 1, pp. 191–204, Feb. 2018.
[49] A. Pérez, P. Larrañaga, and I. Inza, "Supervised classification with conditional Gaussian networks: Increasing the structure complexity from naive Bayes," *Int. J. Approx. Reasoning*, vol. 43, no. 1, pp. 1–25, Sep. 2006.

[50] M. Xu, P. Watanachaturaporn, P. Varshney, and M. Arora, "Decision tree regression for soft classification of remote sensing data," *Remote Sens. Environ.*, vol. 97, no. 3, pp. 322–336, Aug. 2005.
[51] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[52] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Apr. 2006.
[53] T. Hastie, R. J. Tibshirani, and J. Friedman. (2013). *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. [Online]. Available: http://catalog.lib.kyushu-u.ac.jp/ja/recordID/1416361
[54] T. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class AdaBoost," *Statist. Interface*, vol. 2, no. 3, pp. 349–360, 2009.
[55] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, M. Li, J. Xie, M. Lin, Y. Geng, Y. Li, and J. Yuan, "Xgboost: Extreme gradient boosting," *R Package*, vol. 1, no. 4, pp. 1–4, Sep. 2015.
[56] A. A. Taha and S. J. Malebary, "An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine," *IEEE Access*, vol. 8, pp. 25579–25587, 2020.
[57] D. J. Hand, "Assessing the performance of classification methods," *Int. Stat. Rev.*, vol. 80, no. 3, pp. 400–414, Dec. 2012.
[58] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," 2020, *arXiv:2010.16061*.
[59] L. Derczynski, "Complementarity, F-score, and NLP evaluation," in *Proc. Lang. Resour. Eval.*, May 2016, pp. 261–266. [Online]. Available: https://www.aclweb.org/anthology/L16-1040.pdf
[60] M. L. McHugh, "Interrater reliability: The Kappa statistic," *Biochemia Medica*, pp. 276–282, 2012, doi: 10.11613/BM.2012.031.
[61] H. Ebbinghaus, "Memory: A contribution to experimental psychology," *Eur. PMC (PubMed Central)*, vol. 20, no. 4, Oct. 2013.
[62] K. Pearson, "III. Mathematical contributions to the theory of evolution.—XII. On a generalised theory of alternative inheritance, with special reference to Mendel's laws," *Phil. Trans. Roy. Soc. London Ser. Containing Papers Math. Phys. Character*, vol. 203, no. 359, pp. 53–86, 1904.

MOHAMED S. ABDALZAHER (Senior Member, IEEE) received the B.Sc. degree (Hons.) in electronics and communications engineering, in 2008, the M.Sc. degree in electronics and communications engineering from Ain Shams University, Cairo, Egypt, in 2012, and the Ph.D. degree from the Electronics and Communications Engineering Department, Egypt-Japan University of Science and Technology, Madinet Borg Al Arab, Egypt, in 2016.

From April 2019 to October 2019, he joined the Center for Japan-Egypt Cooperation in Science and Technology, Kyushu University, where he was a Postdoctoral Researcher. Currently, he is a Postdoctoral Fellow with the American University of Sharjah. He is an Associate Professor with the Seismology Department, National Research Institute of Astronomy and Geophysics, Cairo. He was elected in the list of top 2% highly ranked scientists. His research interests include earthquake engineering, remote sensing, artificial intelligence, wireless networks, network security, and the IoT. He was a recipient of the State Encouragement Award, in 2022. He was also a recipient of the Scientific Excellence Award from NRIAG, in 2023 and 2024. He was a scientific editor and reviewer in numerous scientific journals and a TPC member of many scientific conferences. He organized and participated in various conferences and workshops. He is an Associate Editor of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

ESSAM GHAMRY received the B.Sc., M.Sc., and Ph.D. degrees from Al-Azhar University, Cairo, Egypt, in 1998, 2002, and 2008, respectively. He is currently a Professor with the National Research Institute of Astronomy and Geophysics, Helwan, Egypt. He was a Postdoctoral Associate with the School of Space Research, Kyung Hee University, South Korea, from 2014 to 2015. He was a Visiting Scholar in U.K., Spain, Germany, India, China, and Malaysia. He has published more than 45 international research articles in the field of applied and space geophysics. He has been the Leader and Co-Leader of two international teams with ISSI, Bern, Switzerland, since 2022. The main focus of his interest is on the observational ground magnetic, electric, and space weather data, with topics related to inner magnetosphere and ionosphere using Van Allen Probe, THEMIS, MMS (NASA's missions), Swarm (ESA's mission), Arase (Japanese/Taiwan mission), and CSES (Chinese mission). His research interests include the application of space geophysics, ionospheric studies and irregularities, magnetic storms, substorms, magnetic micro-pulsations, satellite geomagnetism, GPS/GNSS radio occultation technique using metop and COSMIC satellites, earthquake precursors using ground/space observations, and lithospheric modeling using gravity/magnetic data from LEO satellite. He is a member of the Topical Advisory Panel (Topics Board Editor) with *Universe* journal at MDPI.

KHAIRUL ADIB YUSOF received the B.Sc. degree from Universiti Putra Malaysia (UPM), in 2016, and the Ph.D. degree in space science from Universiti Kebangsaan Malaysia (UKM), in 2021. He is currently a Senior Lecturer with the Department of Physics, UPM. He has authored and co-authored several highly reputable articles in indexed journals, and a book, and owns a patent pending. He has also been invited as a referee/reviewer for several national and international journals. His research interests include earthquake precursor, ground and space geomagnetic observations, signal processing, and computer programming (MATLAB).

MOSTAFA SHAABAN (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from Ain Shams University, Cairo, Egypt, in 2004 and 2008, respectively, and the Ph.D. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 2014. Currently, he is an Associate Professor with the Department of Electrical Engineering and the Director of the Energy, Water, and Sustainable Environment Research Center, American University of Sharjah, Sharjah, United Arab Emirates. He is also an Adjunct Associate Professor with the University of Waterloo. He was with the Independent Electricity System Operator (IESO) of Ontario, Canada, where he was involved in developing generation and flow limits for the stable operation of the power system. Also, he worked for Opus One Solutions as a Clean Energy Consultant. He has several publications in international journals and conferences. His research interests include smart grid, renewable DG, distribution system planning, electric vehicles, storage systems, and bulk power system reliability. He serves as an Associate Editor for *IET Smart Grid*, *IET Energy Conversion and Economics*, *Energy Sustainability Section*, and *Sustainability*, MDPI.