MULTIDISCIPLINARY SCIENCE JOURNAL

RESEARCH ARTICLE
Published Online: November 24, 2024
https://doi.org/10.31893/multiscience.2025299

Analysis of anomaly and novelty detection in time series data using machine learning techniques

Himanshu Sinha

[a]Kelley School of Business, Indiana University, Bloomington, Naperville, IL, United States.

Abstract The process of identifying new patterns or peculiarities that exist in the regular time series data is called time series novelty detection or anomaly detection. Although it is one of the most difficult data mining areas, it is gaining popularity due to its quick application to real-world problems. This research proposes a novel way to detect time series novelty using ML algorithms. An usage of suggested ML techniques to find outliers in time series data has increased recently. Using a dataset from Stack Overflow, this research investigates the use of machine learning for anomaly and novelty identification from time series data.The initial data preparations were dealing with missing values, examining tags of datasets and data visualisation through individuals' words and words using the following assessment metrics: MAE=0. 0629, RMSE=0. 089, and MSE=0. 007. The performance of the second-best model, ARIMA, yielded an MAE = 0. 068, RMSE = 0. 0936 and MSE = 0. 008. The lowest accuracy for this task was witnessed with the Decision Tree Regressor since its error rates were the highest. The results confirm the suitability of the Random Forest Regressor in increasing accuracy in time series data with a special focus on novel and abnormal data point detection while emphasising the significance of the model choice.

Keywords: anomaly detection, time series analysis, machine learning models, stack overflow dataset

1. Introduction

Novelty detection, also called anomaly detection, is the computerised identification of infrequent and atypical instances hidden in a massive amount of ordinary data(S. Arora et al., 2024; Bauskar, 2024). One of its most appealing possible applications is using time series. For example, an autonomous monitoring system that can filter the time series from monitoring sensors and detect any unusual results would be valuable in a safety-conscious environment(Goyal, 2024a, 2024b; Sinha, 2024b). Thus, novelty detection is a challenging issue because there is no clear understanding and uncertainty of 'novelty' for a specific system (Bishukarma, 2021; Ma & Perkins, 2003; Rinky Dwivedi, 2016). It is common practice to use novelty detection methods and approaches when data is available by a certain pattern(Sahil Arora & Apoorva Tewari, 2023), sometimes referred to as the "normal" class rather than the "abnormal" class, which includes the new data(R. Arora et al., 2021; Bauskar, 2023; Chandu, 2024a; Ouafae et al., 2021).

Although these tactics have shown usefulness in particular empirical contexts, their underlying assumptions and processes may cause them to fail in others. Some of them were developed with the erroneous assumption that their creators were familiar with the unique conditions or had correct theoretical models of the core problem, which is seldom the case in reality (Ozdagli & Koutsoukos, 2019; Sinha, 2024e; Thomas, 2024; Thomas et al., 2023; Vennemann et al., 2019). However, this approach misses brief, unique patterns mixed in with regular signals. Leveraging machine learning methods presents an intriguing approach to detecting novel patterns(Ouafae et al., 2021; Rath et al., 2022; Sinha, 2024a).

ML techniques that include AI systems are common in experimental investigations. Regression algorithms, using the input characteristics as inputs, forecast the output values from the data supplied into the system(Suyambu, 2023; Thomas, Vummadi et al., 2024; Thomas & Vedi, 2021; Vishwakarma, 2022). In this instance, the outcome is predicated on acquiring model knowledge during training. A few prominent regression methods for novelty identification include Random Forest Regressor, Decision Tree Regressor, and ARIMA using time series datasets. Natural catastrophes, medical risk, intrusion detection, and urban management are just a few of the many fields that have found growing usage for time series anomaly detection (TSAD) in recent years(Darban et al., 2022; Mathur., 2024; Ritesh Tandon and Aniqa Sayed, 2023). Time series datasets are distinct, which is why machine learning models were selected to be used for novelty discovery in these datasets.

The motivation for this work stems from the increasing importance of effective anomaly and novelty detection in time series data, particularly within dynamic and large-scale data environments such as the Stack Overflow platform. Anomaly and novelty detection profoundly impact data inspection, system and data reliability, and user satisfaction in terms of noticing and addressing undesired behaviour and noticing previously unobserved trends in users' behaviours. This research intends to use

 Received: August 12, 2024 | Accepted: November 6, 2024

advanced machine learning methods to construct and compare models' ability to quickly and accurately identify anomalies in time series data. With such techniques when implemented in the big data set, the study will show how machine learning (S. Arora & Khare, 2024; S. A. Pranav Khare, 2024; Rohilla et al., 2020; sahil Arora, 2023) models can be used to support the prediction maintenance, detection of anomalies, and analysis of trends in real applications. The contribution of this work is as follows:

1. To analyse how well several ML models can identify new or rare phenomena in TS datasets. This work trained and evaluated Random Forest Regressor, ARIMA, and Decision Tree Regressor models on a Stack Overflow dataset.

2. To show a generalised approach to preprocessing on time series data. The study also incorporated data pre-processing parts like handling features like null values, tags, tokens and useful figures like unigram and word clouds. These steps prepared the measurements for the models and analysis during their training and assessment.

3. To use these measures to perform a comparative study of the outcomes from the models. Based on MAE, RMSE, and MSE, the models' respective performances were determined. This demonstrated the advantages and disadvantages of each strategy by clearly comparing the accuracy and error rates of each model..

4. To highlight the importance of model selection in time series prediction tasks. By demonstrating that the RFR outperformed the ARIMA and DT models, the research emphasises an important role of selecting appropriate ML models based on specific dataset characteristics and desired outcomes.

The rest of the article is structured as follows: Section I introduces this research area with contributions and motivation, and the II section provides related work on novelty/anomaly detection. Section II describes the suggested system in detail. The findings and evaluations are detailed in section IV. The next step is to demonstrate how the cutting-edge methods compare. Finally, the report finishes with Section V, which discusses future work.

2. Literature Review

Time-series anomaly detection has seen a great deal of innovative work, focusing on generalised anomaly detection and novelty identification. ML and DL approaches are among the many strategies researchers have used in their research centred on novelty detection.

Bradley et al. (2023) explore an innovative method for discovering survival analysis-based network traffic factors that impact novelty identification. Using RF, Bayesian Ridge, and Linear SVR classifiers, the suggested model successfully pinpoints the crucial traits that affect novelty detection, including PSH Flag Count, ACK Flag Count, URG Flag Count, and Down/Up Ratio.

Lo et al. (2023), provide a method of protection that fortifies novelty detectors against hostile instances by influencing their latent space. The proposed method, PrincipaLS, trains the cascade principle components of the latent space sequentially to improve novelty detectors. After thorough testing on eight assaults, five datasets, and seven novelty detectors, PrincipaLS consistently makes novelty detection models more resilient to adversarial attacks.

Wang et al. (2023) proposed a novel unsupervised model for multivariate TSAD, targeting the challenges of sparse and unlabelled abnormal data, as well as high dimensionality in IoT applications. The results on the SMAP dataset show that our proposed model improves the precision and F1 score by 11.3% and 5.48%, respectively, compared with the latest baseline method. On the NAB dataset, the F1 score is increased by 0.47%. On the SWaT dataset, the precision is improved by 0.62%.

Halim et al. (2023) sought to evaluate SGD One Class SVM, Local Outlier Detection, Elliptical Envelope Covariance, and Isolation Forest—four popular techniques for discovering novelties in network intrusion detection systems. On the UNSW-NB15 dataset, they discovered that Isolation Forest performed better than all other algorithms with an F1-score of 0.723. The outcome demonstrates that novelty detection algorithms have difficulties dealing with network-based IDS.

Bertalanič et al. (2024) offered an improved DITEN anomaly detection solution that manages network physical layer disruptions by converting time-series data to VG and GNN. Our computational efficiency is about 5.9 times higher than Hive-Cote2, and they beat it by 2.2 percentage points. A graph-based method is 10% more effective, and the model is 6% better than the top SOTA imaging model. It reduces computational complexity (CC) by 210 and 4 times similarly. Last, demonstrates that the suggested GNN model and gradient changes from the NVG provide insights that may be understood.

Introducing the novel DeepMaly method Hossen et al. (2024), This method offers a useful tool for SHS developers. Using a novel training method on unlabelled pristine features taken from time series data, DeepMaly effortlessly functions unsupervised, distinguishing between seasonal and genuine abnormalities. The model is trained to identify irregularities in real time using a mix of DCNN and LSTM with an impressive 99.72% accuracy rate.

He et al. (2024) indicated mathematical solutions to synthesise anomalies and fake time series to implement models for real-world applications where there is insufficient data. They have developed Probabilistic Outlier Detection (PrOuD), a generic tool for helping domain specialists with time-series analysis by providing them with easy-to-understand detection findings. The suggested approach may identify new abnormalities with high accuracy and speed, according to experimental findings acquired on both synthetic time series and real-world data from solar inverters.

From the literature review section, the scholars have recorded some progress in the TSAD field by applying various methodologies such as the ML and the DL. Nonetheless, a number of research gaps still need to be filled.. Firstly, while many

studies focus on improving detection accuracy using complex fusion methods or novel architectures, there is a need for more comparative evaluations across different datasets and anomaly types to validate their robustness and generalizability. Secondly, existing approaches often target specific types of anomalies or datasets, such as IoT applications or network intrusion detection, leaving gaps in the adaptation and scalability of these methods to broader domains or mixed types of anomalies. Thirdly, the interpretability of anomaly detection models remains a challenge despite some attempts to provide insights through gradient changes or feature importance analyses. Addressing these gaps could enhance novelty and anomaly detection systems' reliability, applicability, and interpretability across time series datasets with ML techniques. Table 1 provides a Comparative analysis of previous novelty/ anomaly detection studies with key points.

Table 1 Comparative analysis of previous studies of novelty/ anomaly detection.

Author(s)	Techniques	Dataset(s)	Results	Limitation(s)	Future Work
Bradley et al., 2023	Cox proportional hazards, Kaplan-Meier estimates, Random Forest, Bayesian Ridge, Linear SVR	Network traffic data	Identified key features for novelty detection; PSH Flag Count, ACK Flag Count, etc.	Limited evaluation of classifiers	Testing with more diverse network environments
Lo et al., 2023	Principal Latent Space (PrincipaLS), PCA on latent space	Eight attacks, five datasets, seven detectors	Enhanced robustness against adversarial examples	Potential scalability issues with high-dimensional data	Extending to different types of novelty detectors
Wang et al., 2023	Unsupervised model for multivariate TSAD	SMAP, NAB, SWaT	Improved precision and F1 score by up to 11.3%	Limited to IoT-based datasets	Application to more diverse multivariate datasets
Halim et al., 2023	SGD One Class SVM, Local Outlier Detection, Elliptical Envelope, Isolation Forest	UNSW-NB15	Isolation Forest achieved F1-score of 0.723	Difficulties with IDS scalability and performance	Hybrid techniques for improved detection
Bertalanič et al., 2024	DITEN anomaly detection using VG, GNN	Network physical layer data	Outperformed Hive-Cote2 by 2.2%, reduced computational complexity	Limited insight into underlying GNN performance	Investigate interpretability for domain experts
Hossen et al., 2024	DeepMaly (LSTM, DCNN)	Time-series data	Achieved 99.72% accuracy in anomaly detection	Unsupervised approach might limit generalizability	Integration with real-time monitoring systems
He et al., 2024	Probabilistic Outlier Detection (PrOuD), Monte Carlo estimation	Artificial time series, photovoltaic inverter data	Accurate detection of synthetic anomalies	Dependence on synthetic data	Application to more varied real-world datasets

3. Methodology

This research proposes an effective ML-based time-series novelty identification technique. The method employed in this study entailed pre-processing and analysing trends of a dataset from stack overflow. The dataset underwent partitioning, resulting in the creation of distinct training and testing sets and preprocessed using time series windowing. Different predictors such as RFR, DTR, and ARIMA model with Walk Forward Validation were implemented and forecasted. The accuracy of these models was determined via the training and test sets MSE, MAE, and RMSE values with the aid of the actual versus predicted trend plots for a robust trend prediction. These processes are shown in Figure 1, the proposed methodology diagram for the proposed system. The proposed flowchart outlines the methodology for each phase discussed below.

3.1. Data Collection

Data collection is important in research as it creates a premise for subsequent studies(Thomas et al., 2024). The dataset utilised in this study was downloaded from the GitHub repository titled “Trend Prediction on Stack Overflow Dataset” from the GitHub open-source. The data sample contains fields from Stack Overflow, such as tags, scores, and time stamps.

3.2. Data Preprocessing

Data pre-processing is needed to obtain acceptable category classification while evaluating and selecting superior ML algorithms(Bauskar, 2022; Gopalsamy, 2024; Mani Gopalsamy, 2022). This process is not only about normalising the data into similar dynamic ranges and handling missing data and data from different sources but also includes removing outliers and noise and discretising data (D. Singh & Singh, 2020):

Checking Null Values: The first step was to validate the data and check for any absence of data or any Null values. Dealing with missing data is common and crucial to get the quality of the data used for analysis and modelling.

Tag Analysis: The dataset contains tags belonging to several categories that denote diverse issues. In this section, we propose a method of viewing the different tags used in a dataset and then determining their frequency.

Most Common Tags: After that, we displayed the 10 most frequent tags to understand their importance and define the hot subjects in our dataset.

Tag Assignment: A certain value was assigned for each tag column in the given dataset to allow for subsequent analysis.

Data Cleaning: The data cleaning procedure adopted included using the tokenisation technique to break down the textual data into words, excluding the punctuation mark, and converting all the textual data to lowercase. For more textual analysis, the units of analysis were reduced to the unigrams, which showed the most predominantly used words, and a word frequency cloud was produced to show the relative frequency of the appearance of words in the text data.

Windowing: In this study, a sequence length of 10 was selected for the windowing technique to capture an appropriate amount of historical information while maintaining computational efficiency. Testing multiple sequence lengths and comparing their impact on model performance would further validate an effectiveness of this choice.

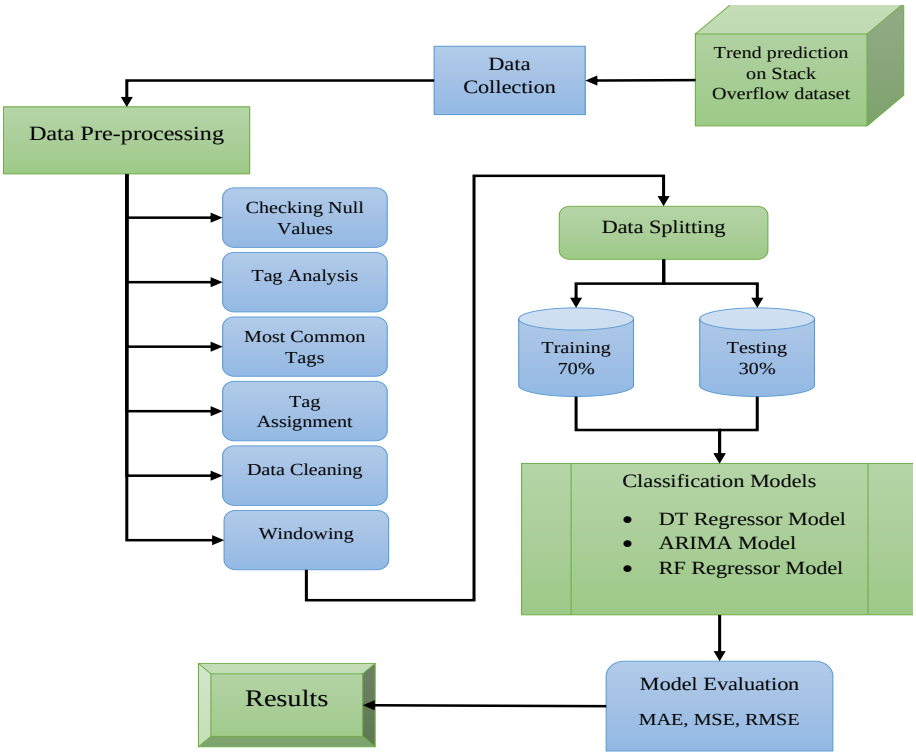

Figure 1 Methodology Diagram of proposed Approach.

3.3. Data Scaling

Feature scaling is one approach for standardising data characteristics or independent variables(Mani Gopalsamy, 2021). Data normalisation is a term used in data processing, often used during data preparation. This inquiry needs to use the min-max normalisation technique. The data was standardised by using Equation 1 as follows:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \qquad (1)$$

X max represents the maximum value in a feature column, and X min represents the lowest value.

3.4. Data Splitting

When validating data, it is common practice to divide it into two sets: training and testing. The dataset was divided into two halves, the training and testing sets, employing a 70:30 split. Therefore, a training set, which is in charge of training the machine learning models, was given 70% of the total data. Thirty percent more of the data was set aside for testing.

3.5. Machine Learning Models

This section outlines the ML models that will be used to implement Time Series Novelty Detection.

3.5.1. Random Forest Regressor Model

The regression technique known as RFR is based on ML. Classification and regression tree (CART) models are the foundation of RF regression, a non-parametric ensemble technique (Breiman, 2001; Rohilla, Chakraborty, et al., 2019; Thota & Arora, 2024a, 2024b). The random subspace and bagging techniques provide its basis(Sinha, 2024c). Bagging and ensembleing learning trees achieves the overall prediction. Learner trees are trained using a series of separate bootstrap samples generated from the initial training data(A. P. A. Singh, 2022; Sinha, 2024d, 2024a). The original training data D, which contains N samples, is used to construct each bootstrap sample (Db). You may swap out some examples when making the bootstrap samples. Typically, Db is around two-thirds of D and doesn't include any duplicate instances. For each set of bootstrap samples, an independent regression tree with an input vector of size b is constructed. Regression trees are known for their high variance and low bias. For regression tasks, an RF prediction is obtained by calculating a mean forecasting of K regression trees, $hk(x)$, as shown in Equation 2 (Ganesh et al., 2021; Merilinna, 2023; Tandon, 2024a; Tyralis & Papacharalampous, 2017; Zhou et al., 2015).

$$RFR\ prediction = \frac{1}{K}\sum_{K=1}^{K} h_k(x) \qquad (2)$$

Regarding the ensemble RFR model, bagging prevents overfitting and reduces variance. The term "bagging" describes a method of producing training data as it is created by randomly resampling. Put another way, rather than eliminating data chosen from input samples; training data are produced while constructing the subsequent subset of data. The learner trees, thus, cannot be connected (Khare, 2023; S. S. Pranav Khare, 2023; Universit et al., 2012).

3.5.2. ARIMA Model

Time-series forecasting is frequently conducted using the ARIMA model (Khan & Alghulaiakh, 2020), which offers a framework for examining stationary data's basic principles and characteristics (R. Arora et al., 2024; Banu et al., 2021; Gopalsamy, 2020; Khare et al., 2024; A. P. A. Singh & Gameti, 2024; Tandon, 2024b). In the generic model known as ARIMA, the variables p (an order of autoregression), d (integration), and q (a moving-average) make up the model's nomenclature. Past actual values and random shocks from a time series, which is a linear function (Hoare et al., 2002; Jambhulkar, 2013; Rohilla, Chakraborty, & Kaur, 2022; Rohilla, Chakraborty, & Kumar, 2022; Tandon, 2024a). The first-order auto-regressive process, denoted as ARIMA (1,0,0) or simply AR (3), is supplied by, for instance, when a time series process {Yi} is provided

$$Y_i = \mu + \varphi_1 Y_{i-1} + \varepsilon_t \qquad (3)$$

And first-order moving average procedure, represented as ARIMA (0,0,1) or just MA (4), is provided by

$$Y_i = \mu - \theta_1 \varepsilon_{i-1} + \varepsilon_t \qquad (4)$$

Another possibility is that the final model is a hybrid of these processes with others of higher levels. Equation 5, therefore, defines a stationary ARMA (p, q) process.

$$Y_i = \varphi_1 Y_{i-1} + \varphi 2 Y_{i-2} + \cdots + \varphi_p Y_{i-p} - \theta_1 \epsilon_{i-1} - \theta_2 \epsilon_{i-2} + \cdots - \theta_q \epsilon_{i-q} + \epsilon_q \qquad (5)$$

Where ε_i' s has a mean of zero and a variance of σ 2 for t = 1, 2,..., n, and is normally distributed independently. It should be noted that the values of p and q typically fall within the range of 0 to 3. The degree of differentiation of the primary variable, Yi.

3.5.3. Decision Tree Regressor Model

Among the many supervised ML structures, DT stand out for their simplicity and utility in making judgements using a tree-like model. Decision trees construct tree-like structures to solve regression or classification issues. In the time series dataset, forecasting decision tree works (Aguilar et al., 2023; Chandu, 2024b; Spiliotis, 2022). As it does so, it progressively builds a decision tree that uses the data it extracts from smaller chunks of data that have comparable values (Gupta, 2024; Kumar et al., 2022; Milunovich, 2020; Rohilla, Kumar, et al., 2019). A tree containing both internal and external branches, as well as leaf nodes, is the final result. One of the input variables (features) accessible at a given node in the tree is contained inside an internal node.

A prediction for a time series value yt using a Decision Tree Regressor can be represented as Equ. 6:

$$\hat{y}_t = f(X_t) \qquad (6)$$

Where:

$\hat{y}_t$ is a forecasted value at time t.

f Represents the decision tree model.

X_t is the feature vector at time t.

The residual r_t at time t is then calculated as, Equ.7.:

$$r_t = y_t - \widehat{y_t} \quad (7)$$

where,
y_t is a real value at time t.

3.6. Model Selection Justification

Therefore, the Random Forest Regressor, ARIMA, and Decision Tree Regressor were selected because each of them provides effective solutions for distinct aspects of the time series data. RFR has shown strong resistance to noise and the issues of overfitting. At the same time, ARIMA is capable of capturing temporal dependencies, and thus, it was used in the study of DTR because of its simplicity and interpretability. Both models have been applied in other research works in the same manner in which they were applied for analysing the time series data for forecasting. Combining these models allows for a comparative evaluation to determine the most effective method for time series novelty detection in the Stack Overflow dataset.

3.7. Model evaluation

The performance of regression models is evaluated based on experimental results using measures such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE).

3.7.1. Mean Squared Error (MSE)

The MSE includes the difference between prediction and the actual values as a measure of performance statistics. You may figure it out using Equation (8).

$$MSE = \frac{1}{N} = \sum(Y - Y')^W \quad (8)$$

3.7.2. Root Mean Squared Error (RMSE)

The RMSE measures the average distance of the difference between forecasted and actual values to give an average of their separation. To calculate RMSE, one uses Equation (9).

$$RMSE = \sqrt{MSE} \quad (9)$$

3.7.3. Mean Absolute Error (MAE)

The MAE was used to calculate the average forecast error from the implemented methodologies. Equation (10) computes it.

$$MAE = \frac{1}{N} = \sum|Y - Y'| \quad (10)$$

In this case, n represents the number of observations, y denotes the target variable's actual value, and y' represents its anticipated value.

4. Results & Discussions

This section describes the proposed research methods through exploratory data analysis (EDA) and experimental results in terms of prediction graph and RMSE, MSE and MAE measures. System requirements include an Intel i7 CPU, 16 GB of RAM (32 GB is ideal), and 512 GB of SSD storage (1 TB is ideal), all running Windows 10 or 11 Pro. To build innovative event detection using ML methods for time series on the Stack Overflow dataset, one must have the following software and hardware configurations: NumPy, Keras, Google Colab, and Pandas. The following sections provide the results of this work.

4.1. Dataset Description

The dataset, sourced from Stack Overflow, comprises 32,890 rows and 10 columns, capturing various aspects of questions asked between April 1, 2022, and May 31, 2022. The features include tags, question ID, owner, creation date, title, score, last activity date, body, and view count. There are 18,669 unique tags in the dataset, with the score column reflecting user votes indicating question helpfulness. This dataset has been used for trend prediction and has undergone extensive preprocessing, including handling tags, tokenising text, scaling features with MinMaxScaler, and creating sequences for model training.

4.2. Exploratory Data Analysis

Exploratory data analysis (EDA) is an integral part of any research data analysis procedure. The main concerns are distributional shape, outlier presence, central tendency (mean, mode, median), and spread (standard deviation, variance) (Sahoo et al., 2019).

Figure 2 shows a bar chart of the first 10 tags of a dataset where ‘python’ was used about 2000 times. Then come tags like 'pandas' and 'dataframe', which have 800 and 500 frequencies, respectively. Other top-10 tags include 'python-3.x', 'Django', 'tkinter','matplotlib', 'numpy', 'list', and 'pygame', which occur less often. The visualisation highlights Python-related inquiries as the dataset's main subjects and trends. To analyse Stack Overflow community interests and guide resource allocation, content production, and user assistance.

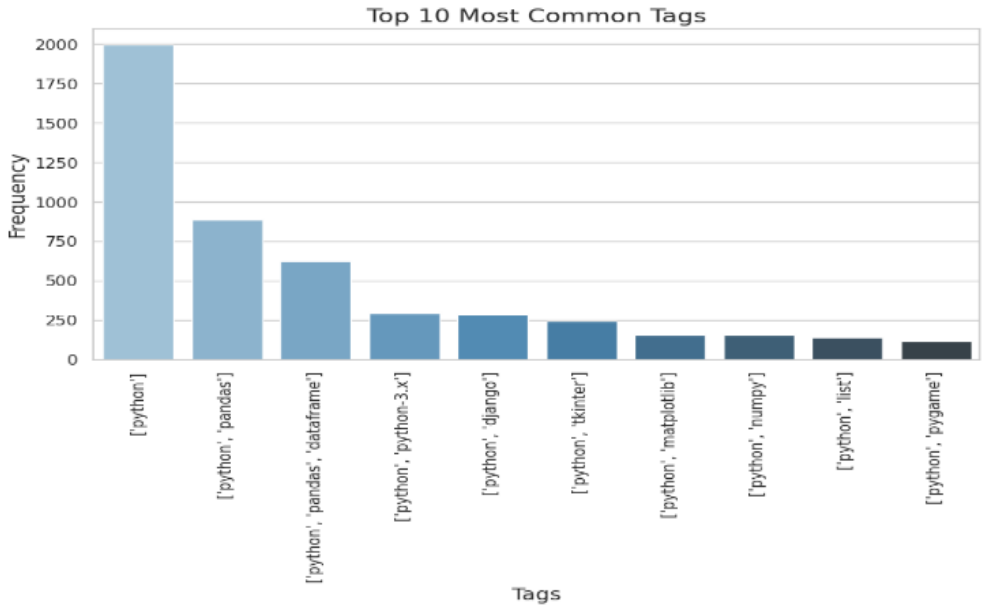

Figure 2 Bar Graph of Tags in Data

The graph in Figure 3 illustrates prevalent unigrams in the dataset in the bar chart. "Python" is the most common term, followed by 'pip' and 'tox'. Other prominent unigrams include 'easyocr','reduce', 'number', and 'jit'. This visualisation identifies the most prevalent words in text data, revealing major subjects and terminology. Knowledge unigram frequencies help feature selection, text analysis, and language model construction by highlighting key terms that may affect the model's predictions and dataset knowledge.

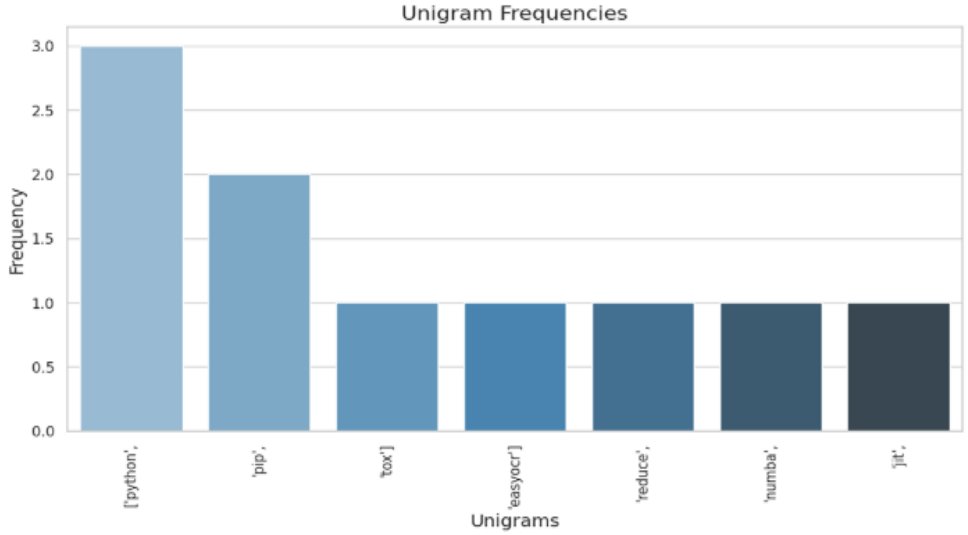

Figure 3 Unigram Graph of Dataset.

The word cloud graph of Figure 4 is a visualisation technique used to depict how often words appear in a text data set. In the specific cloud you sent, the larger the word appears, the more frequently it appears in the text. Words like "python", "pandas", and "machine learning" are all relatively prominent, indicating that the text data centred around these topics.

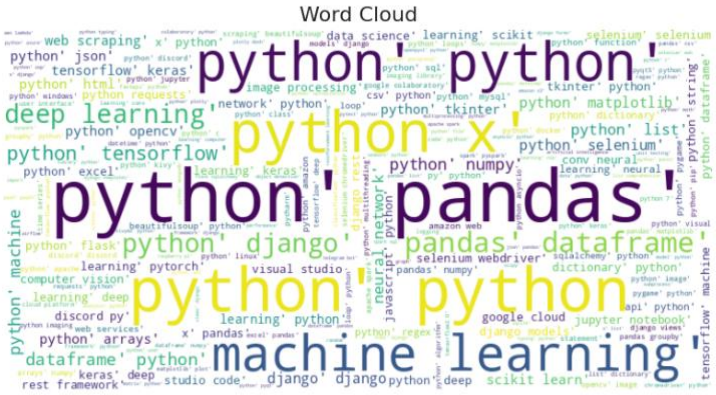

Figure 4 Word Cloud Graph**.**

A bar graph in Figure 5 displays a distribution of values in a dataset labelled "score". The x-axis displays the different values scores can take, and the y-axis displays how many data points have that particular score. For instance, there seems to be a higher concentration of scores around 100 and fewer scores around 200. This suggests that the data may be centred around a value of 100.

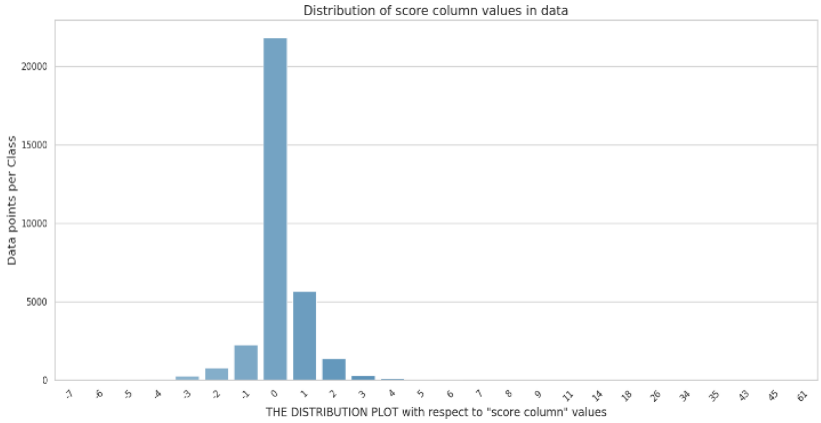

Figure 5 Distribution of Score Counts in Data

Figure 6 depicts the number of inquiries asked with the Python tag over time. The y-axis represents several questions asked, and the x-axis illustrates the time in hours. The number of questions asked appears to follow a cyclical pattern, with peaks occurring every few hundred hours. However, the actual number of questions asked cannot be determined due to the lack of a scale on the y-axis.

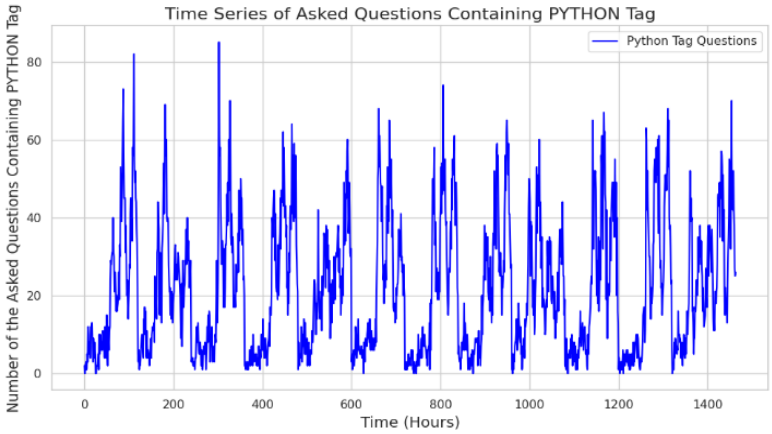

Figure 6 Time Series of Questions Containing Python Tag

4.3. Experimental Results

This section presents experimental findings of a suggested research approach on trend prediction using Machine learning models named RFR, DT, and Arima.

4.3.1. Random Forest Regressor Model

RFR involves creating and training many decision trees using randomly selected feature sets. The average forecast from each tree is used to arrive at the final forecast. The parameters used in this are n_estimators=100 and random_state=42.

The graph in Figure 7 compares predicted values from a random forest model to actual values. The y-axis shows the value, and the x-axis displays an index. It appears actual values are consistently higher than the predicted values. This could indicate that the RF model is underestimating the actual values.

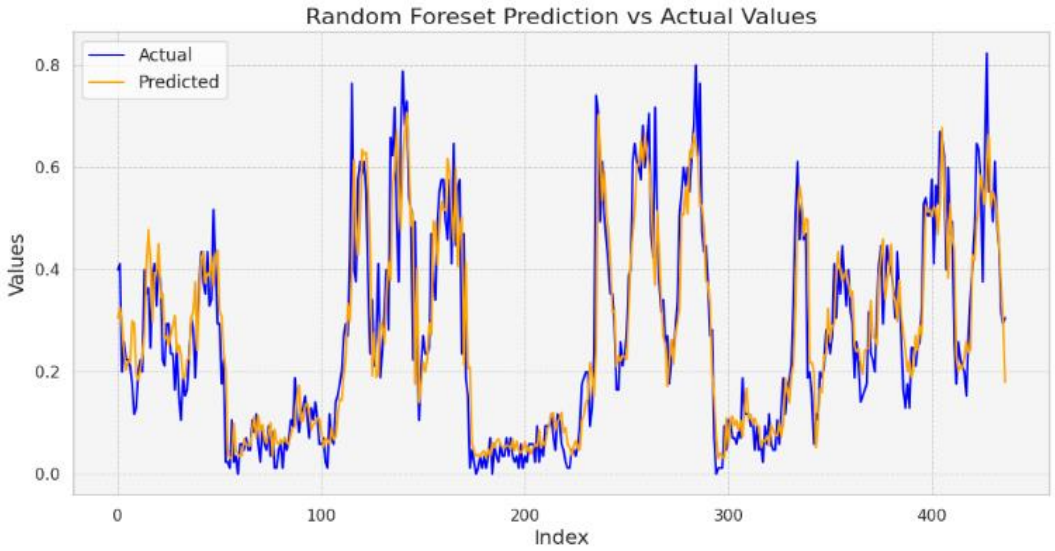

Figure 7 Random Forest Prediction vs Actual Values

An above Figure 8 shows a testing result of a proposed Randon Forest Regressor Model along with the performance measures values of MAE 0.0629, RMSE 0.089, and MSE 0.007.

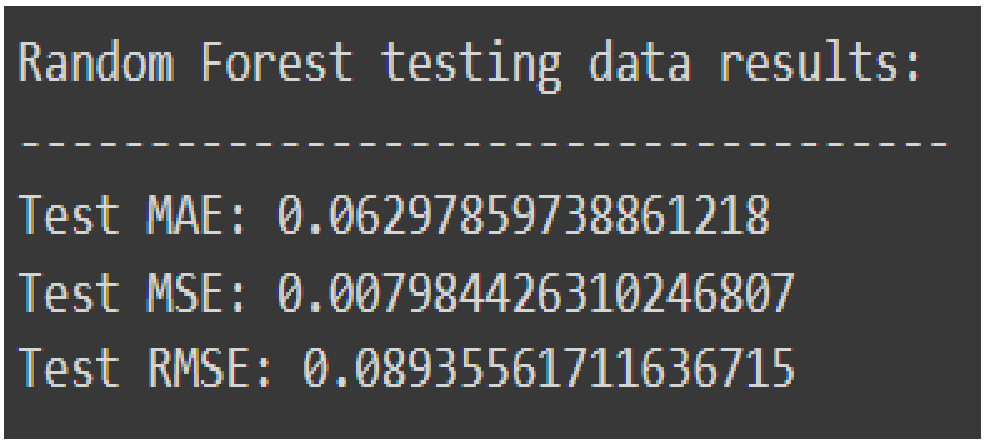

Figure 8 Test Result of Random Forest

4.4. Arima Model

ARIMA forecasts time series data statistically. The ARIMA model with parameters (p=2, d=0, q=4) predicted time series values using Walk Forward Validation. This model utilises the two most recent lag observations (p=2) and the four most recent error terms (q=4).

Graph 9 compares forecasted values (blue line) from an ARIMA model to actual values (red line). A y-axis illustrates a value, and the x-axis displays an index or period. In the specific graph, the predicted values are mostly higher than the actual values, which means the ARIMA model might overestimate the actual values.

Figure 9 compares forecasted values (blue line) from an ARIMA model to actual values (red line). A y-axis illustrates a value, and the x-axis displays an index or period. In the specific graph, the predicted values are mostly higher than the actual values, which means the ARIMA model might overestimate the actual values.

Figure 10 shows MAE0.068, MSE0.008 and RMSE0.0936, which measure the quality of the predictions. Lower values indicate better model performance.

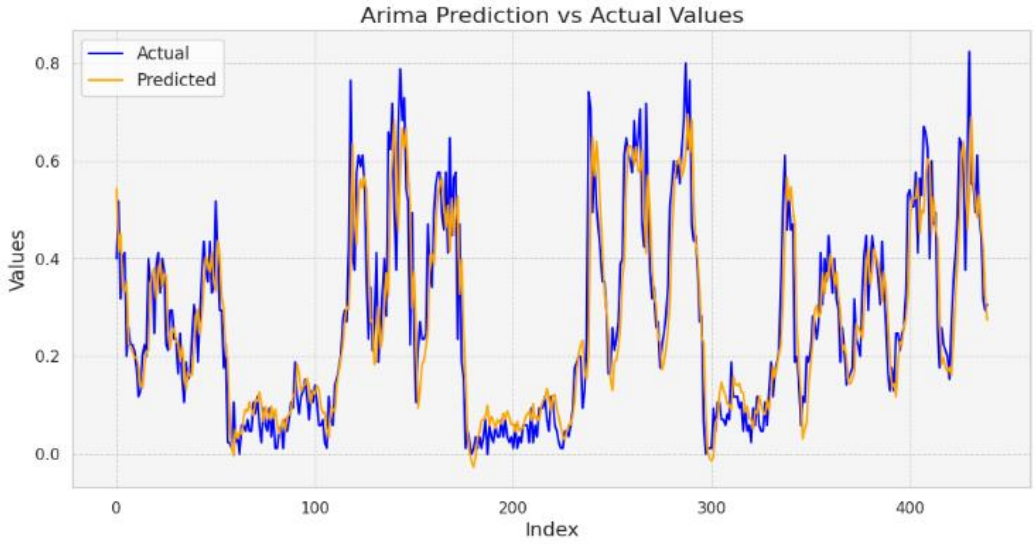

Figure 9 Test Result of Random Forest.

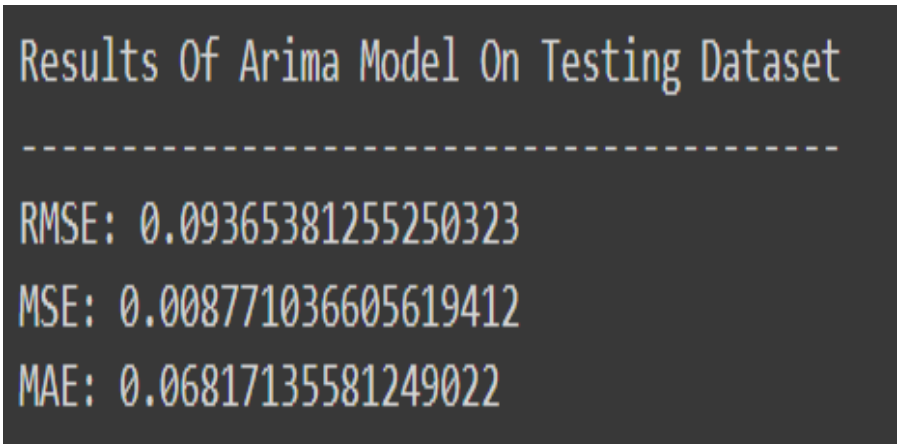

Figure 10 Test Result of Arima Model.

4.5. Decision Tree Regressor Model

In decision tree regression, multiple trees are generated and trained on random subsets of features. The parameters used in this are n_estimators=100 and random_state=42.

The graph in Figure 11 shows the difference between forecasted values by a DT model and the actual values. An x-axis displays the index, and a y-axis shows a value. The graph's predicted values (blue line) are consistently higher than the actual values (red line). Suggesting that the DT model is overestimating the actual values.

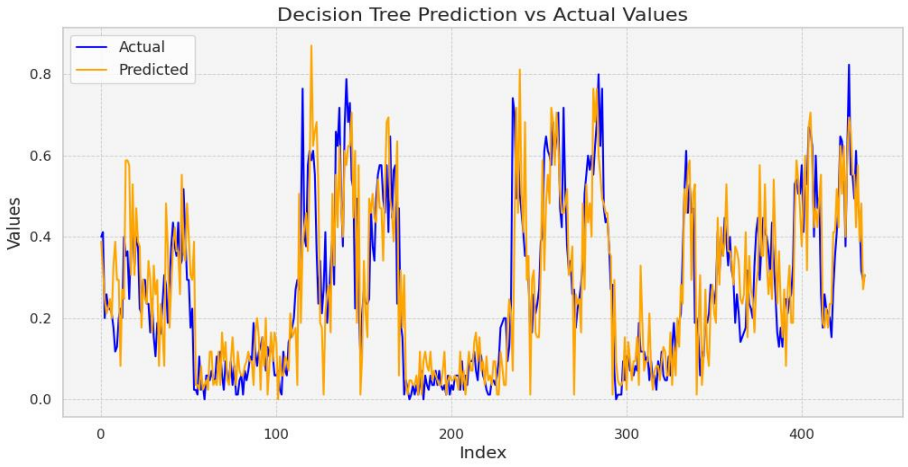

Figure 11 Decision Tree Prediction vs Actual Values

Figure 12 above displays the MSE 0.0156, MAE 0.090 and RMSE 0.125, which serve as metrics for evaluating forecast accuracy. Smaller numbers imply the model's superior performance.

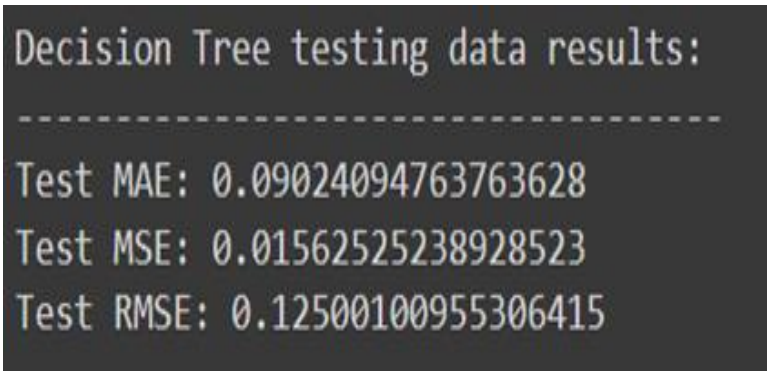

Figure 12 Testing Result of Decision Tree.

4.6. Comparative Analysis and Discussion

Here, the study compares and contrasts the results obtained by the RFR, DTR, and Arima Model of Machine Learning models. The presented models for novelty detection are compared in Table 2 below.

Table 2 Comparison of Machine Learning Models.

Model	Decision Tree Regressor	Random Forest Regressor	Arima Model	LSTM (Guven & Uysal, 2023)	CNN Hybrid (Guven & Uysal, 2023)
MAE	0.090	0.0629	0.0936	5.3750	5.389
MSE	0.015	0.0079	0.0087	59.87	62.42

The study evaluates the performance of various ML models—DTR, RFR, and ARIMA—along with LSTM and CNN Hybrid models for anomaly detection and trend prediction on a Stack Overflow dataset. In terms of Mean Absolute Error (MAE), the Random Forest Regressor (0.0629) outperforms the Decision Tree Regressor (0.090) and ARIMA (0.0936), suggesting it provides more accurate predictions with lower average error. For Mean Squared Error (MSE), the Random Forest Regressor again exhibits superior performance (0.0079), followed by ARIMA (0.0087) and Decision Tree Regressor (0.015), indicating it has the lowest variance of prediction errors. The LSTM and CNN Hybrid models, despite being advanced deep learning approaches, show significantly higher MAE (5.3750 and 5.389 respectively) and MSE (59.87 and 62.42 respectively), implying less effectiveness in this context compared to the simpler Random Forest and ARIMA models. These outcomes highlight the RFR as a most reliable model for this specific anomaly detection and trend prediction task on the Stack Overflow dataset.

5. Discussion

In this study, various machine learning models—DTR, RFR, and ARIMA—were evaluated for anomaly detection and trend prediction on a Stack Overflow dataset. LSTM and CNN Hybrid models were included for comparison. The accuracy of the RFR model was better than that of other models evaluated at MAE 0.0629 and MSE 0.0079, which confirmed the stationary RFR predictions. The forecast of ARIMA was quite good but the values were slightly over-forecasted and the information was also slightly in-built in the analysis showing higher MAE (0.0936) and MSE (0.0087). The Decision Tree Regressor exhibited the highest error among the simpler models, with an MAE of 0.090 and MSE of 0.0156. In this anomaly detection and trend prediction task, other deep learning models, such as LSTM and CNN Hybrid, had higher error values, and thus, it can be concluded that simpler models of machine learning should be preferred for these types of problems.

From an ethical point of view, the use of these models in detecting particular anomalies have questions that deserves an answer such as bias. The complements of this research include ethical implications and impact of AI in anomaly detection. This is especially the case specifically when we try to account for possible biases and probability of Type I and II errors. Even though model performance is a central aspect of the study, there is a problem when biases in the dataset or model result in biased predictions that can unintentionally distort existing trends or patterns. They include false positives where normal behavior is considered as the anomaly and could cause user intercessions to normal activities. On the other hand, false negatives that are failures to detect the true anomalies could potentially enable other damaging trends or problems to be overlooked.

These moral considerations underline the transparency and fairness of the development of the AI algorithm as well as how the resultant models will be implemented. Any real-world use of these models has to take into account such external factors as protection from discrimination and hostile actions against users based on the outcomes of model calculations. There is a constant necessity for model checking against validated models and adapting the criteria for selection of possible biases to

exclude to maintain a balance between technical functionality of the anomalous detection system and its ethical appropriateness.

6. Conclusion and Future Work

Novelty detection uses normal system data to build a model of normality. This research proposed a novel technique for time series novelty identification using ML approaches. The RFR outperformed the other ML models in comparing anomaly and novelty detection using Stack Overflow time series data. It obtained an MAE0.0629, RMSE0.089, and MSE0.007. The model addressed the variability in the dataset well while making slightly conservative actual value estimates. This work shows that the application of the ARIMA model was of moderate accuracy but that it overestimated the actual values with an MAE of 0.068, RMSE of 0.0936 and MSE of 0.008. The Decision Tree Regressor, which had the highest error rates (MAE of 0.090, RMSE of 0.125, and MSE of 0.0156) was less accurate for this time series prediction task. These results highlight the need to identify proper ML models for performing anomaly & novelty detection for higher prediction accuracy. The favourable results of the proposed approach imply its applicability to various real-life large-scale applications in the future, including financial services, production lines, cybersecurity, etc. That is why identifying the possibility of the relationship between the value and the level of confidence in the observed novelty is desirable for further research.

Future work could also investigate increasing model performance using feature engineering similar to this approach but of a higher level, employing deep learning-based structures with more complex patterns or including ensemble learning techniques for accurate identification of anomalous patterns and long-term trends.- Moreover, expanding the problem domain to real-time anomaly detection together with experimental investigations in different fields other than programming could improve the usefulness of these predictive models even more.

Ethical Considerations

Despite the fact that this piece of work do not involve human beings, it becomes appropriate to discuss the issues of ethical considerations in the use of AI for anomaly detection issues such as bias issues and issues that arise from true negatives as well as true positives. Failure to handle biases in data can greatly decrease the chances of correct anomaly detection, which can result in mere interference or major problems being overlooked. To avoid the negative, and reinforcement of bias, intentional and intentional measures, which would guarantee fairness and accountability in model design should be adopted in order to create believable as well as more reliable AI systems.

Conflict of Interest

The authors declare no conflicts of interest.

Funding

This research did not receive any financial support.

References

Aguilar, D. L., Medina-Perez, M. A., Loyola-Gonzalez, O., Choo, K. K. R., & Bucheli-Susarrey, E. (2023). Towards an Interpretable Autoencoder: A Decision-Tree-Based Autoencoder and its Application in Anomaly Detection. *IEEE Transactions on Dependable and Secure Computing*. https://doi.org/10.1109/TDSC.2022.3148331

Arora, R., Gera, S., & Saxena, M. (2021). Mitigating Security Risks on Privacy of Sensitive Data used in Cloud-based ERP Applications. *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 458–463.

Arora, R., Kumar, A., & Soni, A. (2024). *Deep Learning Approaches for Enhanced Kidney Segmentation: Evaluating U-Net and Attention U-Net with Cross-Entropy and Focal Loss Functions*. https://doi.org/10.20944/preprints202408.1816.v1

Arora, S., & Khare, P. (2024). THE IMPACT OF MACHINE LEARNING AND AI ON ENHANCING RISK-BASED IDENTITY VERIFICATION PROCESSES. *International Research Journal of Modernization in Engineering Technology and Science*, *06*(05), 8246–8255.

Arora, S., Khare, P., & Gupta, S. (2024). AI-Driven DDoS Mitigation at the Edge: Leveraging Machine Learning for Real-Time Threat Detection and Response. *2024 International Conference on Data Science and Network Security (ICDSNS)*, 1–7. https://doi.org/10.1109/ICDSNS62112.2024.10690930

Banu, D. P. S., Mary, M. I. A. A., Banu, D. P. S., & Mary, M. I. A. A. (2021). Prediction and Forecasting of Copper Prices using ARIMA models. *Jetir*, *8*(4), 286–290.

Bauskar, S. (2022). BUSINESS ANALYTICS IN ENTERPRISE SYSTEM BASED ON APPLICATION OF ARTIFICIAL INTELLIGENCE. *International Research Journal of Modernization in Engineering Technology and Science*, *04*(01), 1861–1870. https://doi.org/DOI : https://www.doi.org/10.56726/IRJMETS18127

Bauskar, S. (2023). Advanced Encryption Techniques For Enhancing Data Security In Cloud Computing Environment. *International Research Journal of Modernization in Engineering Technology and Science*, *05*(10), 3328–3339. https://doi.org/: https://www.doi.org/10.56726/IRJMETS45283

Bauskar, S. (2024). Enhancing System Observability with Machine Learning Techniques for Anomaly Detection. *International Journal of Management, IT & Engineering*, *14*(10), 64–70.

Bertalanič, B., Hribar, J., & Fortuna, C. (2024). Visibility Graph-Based Wireless Anomaly Detection for Digital Twin Edge Networks. *IEEE Open Journal of the Communications Society*, *5*, 3050–3065. https://doi.org/10.1109/OJCOMS.2024.3393853

Bishukarma, R. (2021). The Role of AI in Automated Testing and Monitoring in SaaS Environments. *IJRAR*, *8*(2). https://www.ijrar.org/papers/IJRAR21B2597.pdf

Bradley, T., Alhajjar, E., & Bastian, N. D. (2023). Novelty Detection in Network Traffic: Using Survival Analysis for Feature Identification. *Proceedings - 2023 IEEE*

International Conference on Assured Autonomy, ICAA 2023. https://doi.org/10.1109/ICAA58325.2023.00010

Breiman, L. E. O. (2001). *Random Forests*. 5–32.

Chandu, H. S. (2024a). *Efficient Machine Learning Approaches for Energy Optimization in Smart Grid Systems*. *10*(9).

Chandu, H. S. (2024b). Enhancing Manufacturing Efficiency: Predictive Maintenance Models Utilizing IoT Sensor Data. *IJSART*, *10*(9).

Darban, Z. Z., Webb, G. I., Pan, S., Aggarwal, C. C., & Salehi, M. (2022). *Deep Learning for Time Series Anomaly Detection: A Survey*.

Ganesh, N., Jain, P., Choudhury, A., Dutta, P., Kalita, K., & Barsocchi, P. (2021). Random forest regression-based machine learning model for accurate estimation of fluid flow in curved pipes. *Processes*. https://doi.org/10.3390/pr9112095

Gopalsamy, M. (2020). Artificial Intelligence (AI) Based Internet-of-Things (IoT)-Botnet Attacks Identification Techniques to Enhance Cyber security. *International Journal of Research and Analytical Reviews (IJRAR)*, *7*(4), 414–420.

Gopalsamy, M. (2024). Identification And Classification Of Phishing Emails Based on Machine Learning Techniques To Improvise Cyber security. *IJSART*, *10*(10).

Goyal, R. (2024a). An Effective Machine Learning Based Regression Techniques For Prediction Of Health Insurance Cost. *International Journal of Core Engineering & Management*, *7*(11), 49–60.

Goyal, R. (2024b). EXPLORING THE PERFORMANCE OF MACHINE LEARNING MODELS FOR CLASSIFICATION AND IDENTIFICATION OF FRAUDULENT INSURANCE CLAIMS. *International Journal of Core Engineering & Management*, *7*(10).

Gupta, K. P. and S. (2024). The Impact of Data Quality Assurance Practices in Internet of Things (IoT) Technology. *International Journal of Technical Innovation in Modern Engineering & Science*, *10*(10), 1–8.

Guven, M., & Uysal, F. (2023). Time Series Forecasting Performance of the Novel Deep Learning Algorithms on Stack Overflow Website Data. *Applied Sciences (Switzerland)*. https://doi.org/10.3390/app13084781

Halim, M., Pratomo, B. A., & Jati Santoso, B. (2023). Comparative Analysis of Novelty Detection Algorithms in Network Intrusion Detection Systems. *2023 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation, ICAMIMIA 2023 - Proceedings*. https://doi.org/10.1109/ICAMIMIA60881.2023.10427625

He, Y., Huang, Z., Vogt, S., & Sick, B. (2024). PrOuD: Probabilistic Outlier Detection Solution for Time-Series Analysis of Real-World Photovoltaic Inverters. *Energies*. https://doi.org/10.3390/en17010064

Hoare, S. W., Asbridge, D., & Beatty, P. C. W. (2002). On-line novelty detection for artefact identification in automatic anaesthesia record keeping. *Medical Engineering and Physics*. https://doi.org/10.1016/S1350-4533(02)00146-7

Hossen, M. J., Hoque, J. M. Z., Aziz, N. A. binti A., Ramanathan, T. T., & Raja, J. E. (2024). Unsupervised novelty detection for time series using a deep learning approach. *Heliyon*. https://doi.org/10.1016/j.heliyon.2024.e25394

Jambhulkar, N. N. (2013). *Modeling of Rice Production in Punjab using ARIMA Model*. *8*, 2–3.

Khan, S., & Alghulaiakh, H. (2020). ARIMA model for accurate time series stocks forecasting. *International Journal of Advanced Computer Science and Applications*. https://doi.org/10.14569/IJACSA.2020.0110765

Khare, P. (2023). *Enhancing Security with Voice : A Comprehensive Review of AI-Based Biometric Authentication Systems*. *10*(2), 398–403.

Khare, P., Arora, S., & Gupta, S. (2024). Integration of Artificial Intelligence (AI) and Machine Learning (ML) into Product Roadmap Planning. *2024 First International Conference on Electronics, Communication and Signal Processing (ICECSP)*, 1–6. https://doi.org/10.1109/ICECSP61809.2024.10698502

Kumar, S. G., Sunny, S., Sayed, A., Jyothidasan, A., Nanda, V., Trinity, J., & Namakkal-Soorappan, R. (2022). Chronic Reductive Stress Modifies Ribosomal Proteins in Nrf2 Transgenic Mouse Hearts. *Free Radical Biology and Medicine*, *192*, 73. https://doi.org/10.1016/j.freeradbiomed.2022.10.125

Lo, S. Y., Oza, P., & Patel, V. M. (2023). Adversarially Robust One-Class Novelty Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. https://doi.org/10.1109/TPAMI.2022.3189638

Ma, J., & Perkins, S. (2003). Time-series Novelty Detection Using One-class Support Vector Machines. *Proceedings of the International Joint Conference on Neural Networks*. https://doi.org/10.1109/ijcnn.2003.1223670

Mani Gopalsamy. (2021). Enhanced Cybersecurity for Network Intrusion Detection System Based Artificial Intelligence (AI) Techniques. *International Journal of Advanced Research in Science, Communication and Technology*, *12*(01), 671–681. https://doi.org/10.48175/IJARSCT-2269M

Mani Gopalsamy. (2022). An Optimal Artificial Intelligence (AI) technique for cybersecurity threat detection in IoT Networks. *International Journal of Science and Research Archive*, *7*(2), 661–671. https://doi.org/10.30574/ijsra.2022.7.2.0235

Mathur., S. (2024). Supervised Machine Learning-Based Classification and Prediction of Breast Cancer. *International Journal of Intelligent Systems and Applications in Engineering*, *12(3)*.

Merilinna, J. (2023). *Advanced Uncertainty Quantification and Novelty Detection for Random Forest Models*. https://doi.org/10.5121/csit.2023.131920

Milunovich, G. (2020). Forecasting Australia's real house price index: A comparison of time series and machine learning methods. *Journal of Forecasting*. https://doi.org/10.1002/for.2678

Ouafae, B., Oumaima, L., Mariam, R., & Abdelouahid, L. (2021). Survey on Novelty Detection using Machine Learning Techniques. *Advances in Science, Technology and Engineering Systems Journal*. https://doi.org/10.25046/aj060510

Ozdagli, A. I., & Koutsoukos, X. (2019). Machine learning based novelty detection using modal analysis. *Computer-Aided Civil and Infrastructure Engineering*. https://doi.org/10.1111/mice.12511

Pranav Khare, S. A. (2024). *Predicting Customer Churn in Subscription-Based Enterprises Using Machine Learning*. *May*, 365–377. https://doi.org/10.1007/978-981-99-8438-1_26

Pranav Khare, S. S. (2023). *AI-Powered Fraud Prevention : A Comprehensive Analysis of Machine Learning Applications in Online Transactions*. *10*(12), 491–497.

Rath, A., Das Gupta, A., Rohilla, V., Balyan, A., & Mann, S. (2022). Intelligent Smart Waste Management Using Regression Analysis: An Empirical Study. *Communications in Computer and Information Science*. https://doi.org/10.1007/978-3-031-07012-9_12

Rinky Dwivedi, V. R. (2016). Empowering Agile Method Feature-Driven Development by Extending It in RUP Shell. *Advances in Computer and Computational Sciences: Proceedings of ICCCCS 2016, 1*.

Ritesh Tandon, Aniqa Sayed, M. A. H. (2023). Face mask detection model based on deep CNN technique using AWS. *International Journal of Engineering Research and Applications Www. Ijera. Com*, *13*(05), 12–19.

Rohilla, V., Chakraborty, D. S., & kumar, D. R. (2019). Random Forest with Harmony Search Optimization for Location Based Advertising. *International Journal of Innovative Technology and Exploring Engineering*. https://doi.org/10.35940/ijitee.i7761.078919

Rohilla, V., Chakraborty, S., & Kaur, M. (2022). Artificial Intelligence and Metaheuristic-Based Location-Based Advertising. *Scientific Programming*. https://doi.org/10.1155/2022/7518823

Rohilla, V., Chakraborty, S., & Kumar, R. (2020). Car Auomation Simulator Using Machine Learning. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3566915

Rohilla, V., Chakraborty, S., & Kumar, R. (2022). Deep learning based feature extraction and a bidirectional hybrid optimized model for location based advertising. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-022-12457-3

Rohilla, V., Kumar, M. S. S., Chakraborty, S., & Singh, M. S. (2019). Data Clustering using Bisecting K-Means. *Proceedings - 2019 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS 2019*. https://doi.org/10.1109/ICCCIS48478.2019.8974537

Sahil Arora, & Apoorva Tewari. (2023). Fortifying Critical Infrastructures: Secure Data Management with Edge Computing. *International Journal of Advanced Research in Science, Communication and Technology*, *3*(2), 946–955. https://doi.org/10.48175/IJARSCT-12743E

sahil Arora, P. K. (2023). The Role of Machine Learning in Personalizing User Experiences. *JETIR*, *11*(6), 1–1.

Sahoo, K., Samal, A. K., Pramanik, J., & Pani, S. K. (2019). Exploratory data analysis using python. *International Journal of Innovative Technology and Exploring Engineering*. https://doi.org/10.35940/ijitee.L3591.1081219

Singh, A. P. A. (2022). STRATEGIC APPROACHES TO MATERIALS DATA COLLECTION AND INVENTORY MANAGEMENT. *International Journal of Business Quantitative Economics and Applied Management Research*, *7*(5).

Singh, A. P. A., & Gameti, N. (2024). Leveraging Digital Twins for Predictive Maintenance: Techniques, Challenges, and Application. *IJSART*, *10*(09), 118–128.

Singh, D., & Singh, B. (2020). Investigating the impact of data normalization on classification performance. *Applied Soft Computing*. https://doi.org/10.1016/j.asoc.2019.105524

Sinha, H. (2024a). A Comprehensive Study on Air Quality Detection Using ML Algorithms. *Journal of Emerging Technologies and Innovative Research (JETIR) Www.Jetir.Org*, *11*(9), b116–b122.

Sinha, H. (2024b). Benchmarking Predictive Performance Of Machine Learning Approaches For Accurate Prediction Of Boston House Prices : An In-Depth Analysis. *Ternational Journal of Research and Analytical Reviews (IJRAR)*, *11*(3).

Sinha, H. (2024c). Predicting Bitcoin Prices Using Machine Learning Techniques With Historical Data. *International Journal of Creative Research Thoughts (IJCRT)*, *12*(8). https://doi.org/10.3390/e25050777

Sinha, H. (2024d). Predicting Employee Performance in Business Environments Using Effective Machine Learning Models. *International Journal Of Novel Research And Developmen*, *9*(9), 875–881.

Sinha, H. (2024e). The Identification of Network Intrusions with Generative Artificial Intelligence Approach for Cybersecurity. *Journal of Web Applications and Cyber Security*, *2*(2), 20–29. https://doi.org/10.48001/jowacs.2024.2220-29

Spiliotis, E. (2022). Decision Trees for Time-Series Forecasting. *Foresight: The International Journal of Applied Forecasting*.

Suyambu, P. K. V. and M. R. (2023). A Study on Energy Management Systems (EMS) in Smart Grids Industry. *International Journal of Research and Analytical Reviews (IJRAR)*, *10*(02), 558–563.

Tandon, R. (2024a). An Analysis Of COVID-19 Tweets Sentiments Based On Large Language Models (Llms). *International Journal of Research and Analytical Reviews (IJRAR)*, *11*(3), 319–328.

Tandon, R. (2024b). The Machine Learning Based Regression Models Analysis For House Price Prediction. *International Journal of Research and Analytical Reviews (IJRAR)*, *11*(3), 296–305.

Thomas, J. (2024). *Optimizing Nurse Scheduling : A Supply Chain Approach for Healthcare Institutions*. 2251–2259.

Thomas, J., Vedi, K. V., & Gupta, S. (2023). *An analysis of sustainable e-commerce logistics in supply chain management*.

Thomas, J., Vedi, K. V., & Gupta, S. (2024). Artificial Intelligence and Big Data Analytics for Supply Chain Management. *International Research Journal of Modernization in Engineering Technology and Science*, *06*(09). https://doi.org/DOI : https://www.doi.org/10.56726/IRJMETS61488

Thomas, J., & Vedi, V. (2021). Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *8*(9).

Thomas, J., Vummadi, J., & Shah, R. (2024). *Machine Learning Driven Device for Enhanced Quality Oversight in Supply Chains*.

Thota, S. R., & Arora, S. (2024a). *COLLABORATIVE FILTERING AND KNOWLEDGE GRAPHS FOR DATA DISCOVERY*. *05*, 8679–8692.

Thota, S. R., & Arora, S. (2024b). Neurosymbolic AI for Explainable Recommendations in Frontend UI Design-Bridging the Gap between Data-Driven and Rule-Based Approaches. *International Research Journal of Engineering and Technology*, *11*(5).

Tyralis, H., & Papacharalampous, G. (2017). Variable selection in time series forecasting using random forests. *Algorithms*. https://doi.org/10.3390/a10040114

Universit, L., Curie, M., Bo, P. V. I., Cedex, P., & Yu, B. (2012). *Analysis of a Random Forests Model*. *13*, 1063–1095.

Vennemann, B., Obrist, D., & Rösgen, T. (2019). Automated diagnosis of heart valve degradation using novelty detection algorithms and machine learning. *PLoS ONE*. https://doi.org/10.1371/journal.pone.0222983

Vishwakarma, M. R. S. and P. K. (2022). An Efficient Machine Learning Based Solutions for Renewable Energy System. *International Journal of Research and Analytical Reviews (IJRAR)*, *9*(4), 951–958.

Wang, F., Yan, M., Li, Q., & Wang, C. (2023). A Multivariate Time Series Anomaly Detection Model Based on Spatio-Temporal Dual Features. *Proceedings - 2023 International Conference on Networking and Network Applications, NaNA 2023*. https://doi.org/10.1109/NaNA60121.2023.00075

Zhou, Q. F., Zhou, H., Ning, Y. P., Yang, F., & Li, T. (2015). Two approaches for novelty detection using random forest. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2014.12.028