Enhancing Smart Grid Reliability Through Anomaly Detection and Theft Detection

Honey Patel[1][0009-0001-1561-2013], Yash Rank[2], Smit Surani[3] and Aviral[4]

[1] Parul University, Vadodara, Gujarat, India
[2] Parul University, Vadodara, Gujarat, India
[3] Parul University, Vadodara, Gujarat, India
[4] Parul University, Vadodara, Gujarat, India
200303125009@paruluniversity.ac.in
210303125004@paruluniversity.ac.in
210303125005@paruluniversity.ac.in
210303125016@paruluniversity.ac.in

Abstract. As energy grids become more complex, the need for effective monitoring the grid and security measures for utility services is growing. This research focuses on anomaly and theft detection in smart energy grids using data analytics and machine learning techniques applied on the real-world Transmission and Distribution (T&D) data by Supervisory Control and Data Acquisition (SCADA) data obtained from Madhya Gujarat Vij Company Limited (MGVCL). The methodology proposes an analytical unit for data preprocessing, anomaly detection with single-class classifiers, and forecasting key metrices. Anomaly detection identifies potential mishaps, which could signal faults or illegal usage. Machine learning models are used to predict future energy patterns, helping to improve grid's robustness. Additionally, a dashboard based on Microsoft PowerBI is proposed for real-time visualization of observed patterns. The study also discusses the implementation of smart meters to collect and monitor data in real-time. This approach aims to enhance the security and efficiency of energy grids by providing timely detection of potential issues.

Keywords: Smart Energy Grid, Data Analytics, Anomaly Detection, Machine Leaning

1 Introduction

1.1 Background and Motivation

The energy sector is undergoing significant changes with advanced technologies, leading to the development of smart energy grids. These grids, including smart meters and communication networks, give improved efficiency and real-time monitoring capabilities. Anomalies such as unauthorized energy usage can lead to disruptions and financial losses. Existing methods for detecting these issues often fall short in terms of accuracy and utilize a lot of man power and time as that involves manual visual inspection without any prior knowledge using brute force type of methodology, highlighting the need for more advanced solutions.

1.2 Problem Statement

The predominant challenge in smart energy grids is the real-time detection of anomalies and theft, which can compromise the grid's integrity. There is a pressing need for a proactive approach that can detect these issues in real-time, allowing for timely interventions. Additionally, accurate forecasting of energy consumption and other metrices is essential to optimize grid operations and prevent potential overloads or shortages.

1.3 Objective

We aim to develop an analytical layer on the top of the data available with the electricity substation to detect the anomalies and theft in heuristic manner rather than a completer uninformed visual inspection in the city using data analytic in python and Power BI. The methodology includes, utilizing single-class classifiers to identify outliers or anomalies from normal grid operations, which may indicate faults or unauthorized activities, applying regression models from classical machine learning to predict future patterns, enhancing grid management. Implementing a dashboard for real-time monitoring and visualization of key metrics, providing grid operators with actionable insights. As shown in Fig. 1 the analytical layer will be placed as the pink cube in the diagram. Dotted lines show data transfer and solid lines show physical connection. Objectives
The main objectives of this research are:

- To design an analytical layer on the top of tradition SCADA
- To design a robust analytical pipeline for timely anomaly detection within smart energy grids.
- To implement effective theft detection algorithms to detect unauthorized energy usage.
- To create an accurate forecasting model using machine learning techniques.
- To design a PowerBI dashboard that facilitates real-time monitoring and decision-making.

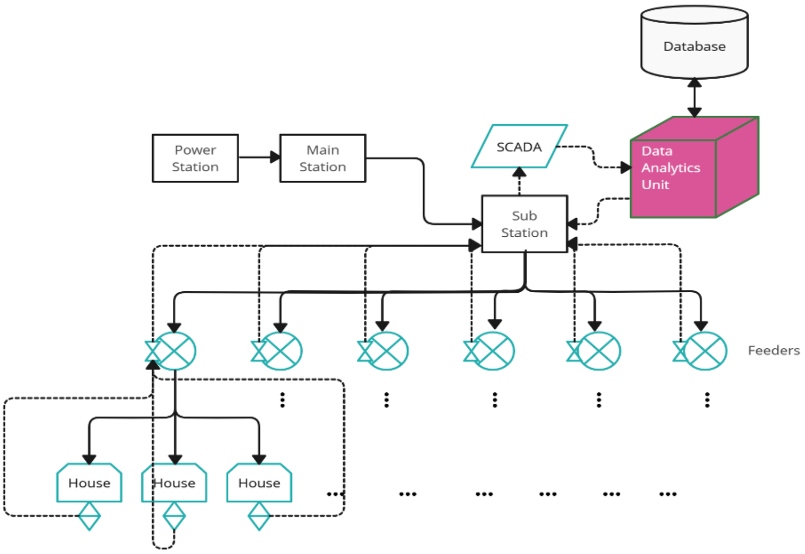

Fig. 1. Schematic diagram of the Transmission and Distribution (T&D) system in India

2 Literature Review

Table 1. Literature Review

Authors	Key Takeaways
Khan, I. U., Javaid, N., Taylor, C. J., & Ma, X. (2023)	Proposed data-driven approaches to combat electricity theft in smart grids.
El-Toukhy, A. T., Badr, M. M., Mahmoud, M., Srivastava, G., Fouda, M., & Alsabaan, M. (2023)	Used deep reinforcement learning to improve theft detection in smart grids.
Abdulaal, M., Ibrahem, M. I., Mahmoud, M., Khalid, J., Aljohani, A., Milyani, A. H., & Abusorrah, A. (2022)	Focused on real-time detection of false readings using ensemble learning in smart grid AMI.
Elgarhy, I., Badr, M. M., Mahmoud, M., Fouda, M. M., Alsabaan, M., & Kholidy, H. A. (2023)	Employed clustering and ensemble methods to secure theft detectors against evasion attacks.
Zheng, K., Chen, Q., Wang, Y., Kang, C., & Xia, Q. (2021)	Combined data-driven approaches to detect electricity theft effectively.
Arif, A., Javaid, N.,	Emphasized big data analytics for

Aldegheishem, A., & Alrajeh, N. (2021)	identifying electricity theft in microgrids.
Lepolesa, L. J., Achari, S., & Cheng, L. (2021)	Utilized deep neural networks for theft detection in smart grids.
Ayub, N., Ali, U., Mustafa, K., Mohsin, S. M., & Aslam, S. (2021)	Introduced predictive data analytics for electricity fraud detection using CNN.
Althobaiti, A., Jindal, A., Marnerides, A. K., & Roedig, U. (2021)	Surveyed data-driven attack strategies and detection methods for theft in smart grids.
Elahe, M. F., Jin, M., & Zeng, P. (2021)	Reviewed load data analytics using deep learning in smart grids.
Ahmed, M., Khan, A., Ahmed, M., Tahir, M., Jeon, G., Fortino, G., & Piccialli, F. (2022)	Analyzed challenges and proposed solutions for energy theft detection.
Takiddin, A., Ismail, M., & Serpedin, E. (2023)	Focused on detecting adversarial evasion attacks in smart grids using robust methods.
Oprea, S. V., & Bâra, A. (2022)	Applied SQL-based feature engineering for detecting electricity fraud using machine learning.

All the papers studied and reviewed during the process have been a great source of inspiration, However, the papers do talk about smart meters, demand forecasting and more but are not very apt Indian context. Countries like United States, European Union and other developed nations are advancing their technology and are retiring SCADA while India is yet to complete installation of SCADA entirely. This gap in technology makes these paper not very apt in our context.

More importantly, our focus revolves around anomaly and theft detection to ensure reliability and efficiency of the grid. There has not been much research in this area from data analytics perspective. This analytical layer is the research gap that we attempt to bridge.

3 Methodology

The methodology for this research is designed to address the challenges of anomaly detection, theft detection, and forecasting within smart energy grids.

3.1 Data Collection

The dataset used in this research was provided by Madhya Gujarat Vij Company Limited (MGVCL) Vadodara for the Gorwa Sub Station. This dataset includes comprehensive information about energy distribution and consumption, focusing on various feeders managed by the substation.

Key features are Columns like "Feeder Name", "Feeder Type", and "Feeder Category". Metrics such as "SS2FEEDER Units(Kwh)", "Feeder2SS Units(Kwh)", and "Total Sentout" represent the energy transmitted through the grid. The dataset also includes "Total Billed Units", "Unite Loss", and "T&D Loss (%)", which are vital for detecting discrepancies and potential anomalies. The "Month" and "Year" columns allow for the analysis of seasonal patterns and trends in energy usage over time.

Understanding the Data

The dataset contains 19 columns and 252 entries. Special attention was given to features like "Unit Loss" and "T&D Loss (%)" as they directly indicate potential issues such as energy theft or transmission losses.

3.2 Data Preprocessing

Data Loading and Transformation

Numerical columns were converted to floats for accurate computation. Key features related to energy flow, such as "SS2FEEDER Unites(Kwh)(8)", "Total Sentout", and "Total Billed Units," were the focus of the analysis.

Data was non-gaussian in nature. The Data was converted to gaussian using Box-Cox, Log Transformation and Quantile Transformation. The data was grouped by "Feeder Name" to perform aggregated analyses. Seasonal trends and variations were examined for different feeders, which is crucial for accurate forecasting.

3.3 Exploratory Data Analysis (EDA)

Summary statistics were computed to understand the central tendency and dispersion of key variables. Distribution plots and histograms were generated for key variables to inspect their distribution and detect any anomalies. Line plots were used to visualize seasonal variations in energy usage across different feeders, providing insights into cyclical patterns. A correlation matrix was created to identify relationships between different energy metrics.

3.4 Anomaly Detection

Single-class classifiers like One-Class SVM or Isolation Forest were trained on normal operational data to detect deviations indicative of anomalies.

This step involved, Training on historical data representing normal grid behaviour

3.5 Forecasting

For time series forecasting traditional Machine Learning models such as Polynomial Regression, Gradient Boosting Regressor, Random Forest Regressor or Decision Tree Regressor were considered based on the temporal and gaussian nature of the data.

Model performance was evaluated using metrics like Root Mean Squared Error (RMSE) and R² scores to ensure accuracy.

3.6 Power BI Dashboard

The Power BI dashboard was designed to display real-time data, including current energy consumption, detected anomalies, and forecasted usage trends. Interactive elements were included to allow users to filter data, zoom into specific periods, and customize the view according to their needs. Fig 2 displays all the modules and steps mentioned in the methodology. It serves as a schematic for the journey of the data and operations performed on the data.

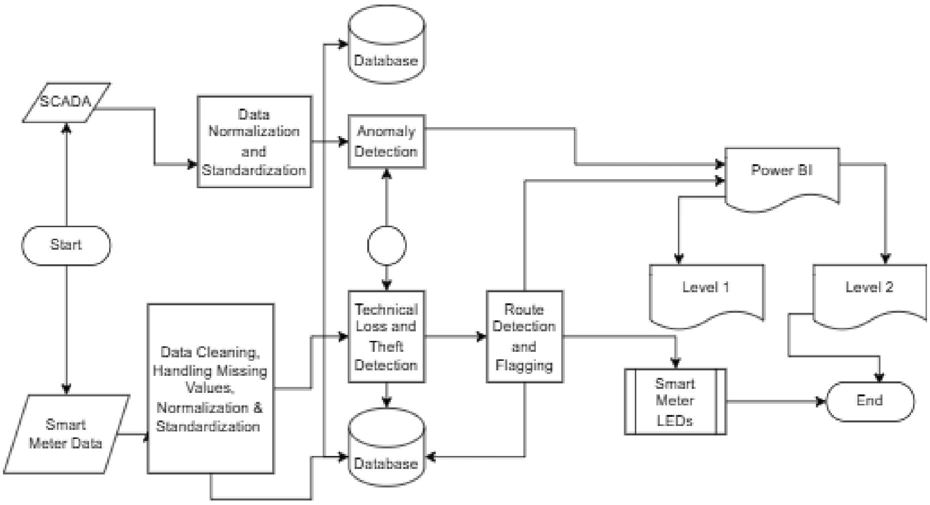
SCADA
Start
Smart Meter Data
Data Normalization and Standardization
Data Cleaning, Handling Missing Values, Normalization & Standardization
Database
Anomaly Detection
Technical Loss and Theft Detection
Database
Route Detection and Flagging
Power BI
Level 1
Level 2
Smart Meter LEDs
End

Fig. 2. Data Pipeline

3.7 System Scalability and adaptability

Proposed algorithms are highly scalable in nature and adaptable in any scenario. Cloud Computing technology can be employed in order to scale the models for implementation for larger grids. Machine learning algorithms that have been employed are Isolation Forest and Random Forest Regressor. Both the algorithms are tree based and ensemble models thus are highly adaptive in nature. These models can be easily be deployed on public or private cloud architectures and are adaptive to all sizes, physiography of the grid and variety of grid operation.

4 Implementation

4.1 Libraries

The necessary libraries, including pandas, numpy, matplotlib, seaborn, scipy, and sklearn, were imported to facilitate data manipulation, statistical analysis, visualization, and machine learning.

4.2 Data Transformation

1. **Box-Cox Transformation**: Applied to columns like 'SS2FEEDER Unites(Kwh)', 'Total Sentout', and others to stabilize variance and normalize the data.
2. **Log Transformation**: Used on the 'Consumer Export(10)' column to handle skewness.
3. **Quantile Transformation**: For the 'HT Sold(15)' column, a combination of log transformation, standardization, and quantile transformation was applied to handle multimodal distribution.

4.3 Label Encoding

Categorical variables were encoded into numerical values to facilitate model training and improve computational efficiency. The 'Month' and ‘Feeder name’ column, originally containing categorical data, was encoded into numerical format.

4.4 Outlier Detection

Outlier detection was implemented using the Isolation Forest algorithm to identify anomalies in the T%D Loss(%)(21=20/11), Total Sentout, LT Sold Out, HT Sold Out, Total Soldout, Unit Loss, SS2FEEDER, FEEDER2SS, Consumer Export, Total Billed, Consumer Imports and Net Assessed columns.

Given that we are dealing with an energy meter and smart grid system, where power fluctuations or abnormal usage patterns could signal anomalies like energy theft or technical losses, we propose using Isolation Forest as this algorithm is well-suited for:

- Complex non-linear data like energy consumption patterns.
- Real-time or large datasets.
- Cases where outliers are important, such as detecting abnormal energy consumption, which could indicate theft.

As the data was transformed into gaussian distribution other methods statistical methods such as Z-Score, IQR and few more were tried. In addition to this single class classification algorithms such as OneClass SVM, DBSCAN, Local Outlier Factor and many more were tried, however due to above mention advantages Isolation Forest outperformed others.

There are total of 21 Feeders in the given data. As T&D loss is recorded in percentages its outlier is detected taking all the feeders in account. However, all other columns go for anomaly detection with respect to the feeder's name column. As values for each feeder must be in the normal distribution of its respective feeder.

In the implementation of the Isolation Forest algorithm, the line `iso_forest = IsolationForest(n_estimators=7, contamination=0.22, random_state=42)` is crucial for initializing the model with specific parameters. The IsolationForest class, sourced from the Scikit-learn library, is used to create an instance of the Isolation Forest algorithm, which is designed for anomaly detection. The parameter n_estimators=7 specifies that the model will construct 7 isolation trees. The choice of 7 trees is a balance between computational efficiency and the robustness of the model.

The parameter contamination=0.22 denotes the expected proportion of outliers within the dataset. This parameter is critical as it informs the model about the expected anomaly rate, which affects the threshold for classifying an instance as an anomaly. By setting the contamination parameter to 22%, the model is configured to anticipate that approximately 22% of the data points are anomalies, thus improving the model's sensitivity and specificity.

Lastly, the random_state=42 parameter ensures that the randomness involved in the model's training process is controlled, allowing for reproducible results. By setting the random state to a fixed value, such as 42, the model's behavior is consistent across different runs with the same data.

4.5 Forecasting

The dataset is prepared and utilized to train and evaluate a Random Forest regression model for predicting the total sentout, total billed and unit loss. For the forecasting module several algorithms were given a try based on the visual inspection of the graph. Algorithms namely Polynomial Regression of degree 2,3 and 4, Support Vector Regressor, Decision Tree Regressor, Random Forest Regressor, K-Neighbours Regressor, Elastic Net and many more were fairly tried and performed a randomized and grid search for hyperparameter tuning but failed to give satisfactory R^2 Score. Other major reason for the selection of Random Forest was its versatile nature as it is an tree based and ensemble model that makes it highly adaptable and scalable. Random Forest despite of being scalable is resistant to overfitting. These were the major reason to adopt Multiple Random Forest Regressor. The features included are the 'Month' and 'Feeder name' columns, which are combined into a single feature matrix x using np.column_stack. The target variable y is the 'Total Sentout', 'Total Billed' and 'Unit Loss' respectively.

To enhance the model, polynomial feature expansion is applied. The PolynomialFeatures class from Scikit-learn is used with a degree of 2, which allows for the generation of interaction terms and polynomial features up to the specified degree.

This transformation is performed on the feature matrix X to create a new feature matrix X_{poly}, which includes both the original features and their polynomial interactions.

The dataset is then split into training and testing subsets using train_test_split, with 20% of the data allocated to the test set. The feature matrix and target variable are scaled using StandardScaler to standardize the data. The training and testing data are scaled separately, ensuring that the model is evaluated on a consistent scale.

Hyperparameter tuning for the Random Forest model is conducted using GridSearchCV. The grid of hyperparameters includes n_estimators (the number of trees in the forest), max_depth (the maximum depth of the trees), min_samples_split (the minimum number of samples required to split an internal node), and min_samples_leaf (the minimum number of samples required to be at a leaf node). In this implementation, the parameter grid includes several values for each hyperparameter: 100, 200, and 300 trees; no limit, 10, 20, and 30 maximum depths; 2, 5, and 10 minimum samples for splitting; and 1, 2, and 4 minimum samples per leaf.

The GridSearchCV function is used to evaluate each combination of parameters through 5-fold cross-validation, optimizing for the R^2 score. The best model, identified by grid_search_rf.best_estimator_, is then used to make predictions on both the training and testing data. The predictions are inverse transformed to their original scale using the scaler_y, and the performance of the model is assessed using mean squared error (MSE) and R^2 score metrics.

The performance of the model is visualized using 3D scatter plots. The plots compare the actual and predicted values of the training and testing datasets. The first subplot displays the training data points and the predictions, while the second subplot shows the testing data.

To analyze the significance of each feature in the Random Forest model, we compute the feature importances using the feature_importances_ attribute of the best_rf_ts model, which represents the best Random Forest estimator obtained from hyperparameter tuning. The feature importances indicate how much each feature contributes to the prediction of the target variable.

Individual Tree Prediction

For each decision tree T_k in the forest, the prediction y_hat is made using the two input variables X_1 and X_2

$$y\_hat(k) = Tk(X1, X2) \tag{1}$$

Averaging Predictions

The Random Forest Regressor aggregates the predictions from all N decision trees by averaging them

$$y\_hat = \frac{1}{N}\sum_{k=1}^{N} y\_hat(k) \tag{2}$$

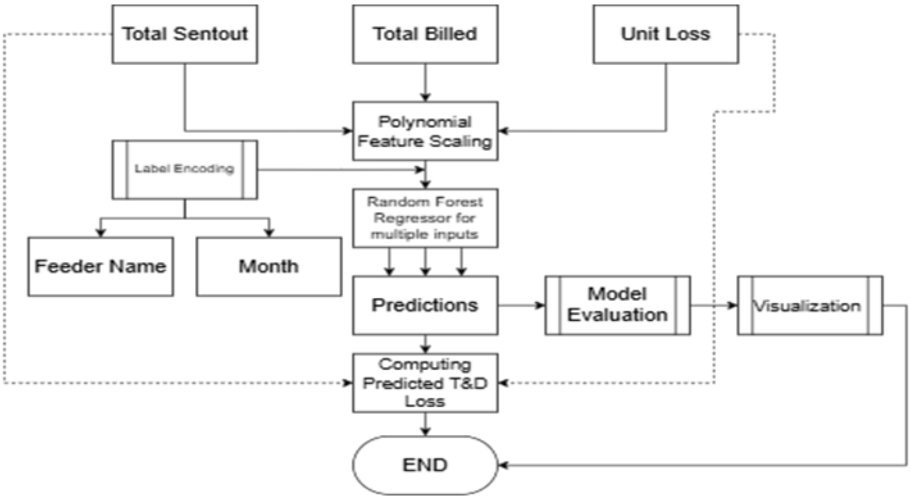

Fig. 3. Steps involved in forecasting

General Formulation

For a general case where you have N decision trees, the final prediction y_hat for input features X_1 and X_2 is:

$$y\_hat \;=\; \frac{1}{N}\sum_{k=1}^{N} T_k(X_1, X_2) \qquad (3)$$

4.6 Power BI Dashboard

Power BI is a data visualization and dashboard creating tool by Microsoft. This creates interactive and informative visuals that can be easily understood by naïve users as well. Power BI offers wide variety of plots, graphs and charts and is also easy to integrate with any data source for real-time data. Power BI can be easily used by the employes to find and detect the anomalies using graphs that clearly show the anomalies in different colour. This helps the department in easy monitoring of the data which is otherwise very difficult. This visual format of data make decision making very fast and efficient.

If the department has trained staff, they can merely hover over a point in plot to get even tiniest detail of it. Power BI offers a lot of formulation features like filtering, grouping, sorting and transforming and many more that are vital in scenario of big data and much useful for the grides to target specific clusters. Proposed dashboard structure is as below:

Level 1: Anomaly Detection and Forecasting

The first level of the Power BI dashboard is designed to provide a comprehensive view of anomaly detection and forecasting. This section serves to analyze historical data to identify anomalies and assess the accuracy of forecasting models.

Anomaly Detection component of the dashboard summarizes the anomalies identified in the dataset. It includes visualizations such as charts that display the frequency and distribution of anomalies over time or across different categories. By examining these visualizations, users can quickly grasp the extent and nature of the anomalies, identifying patterns or trends that warrant further investigation.

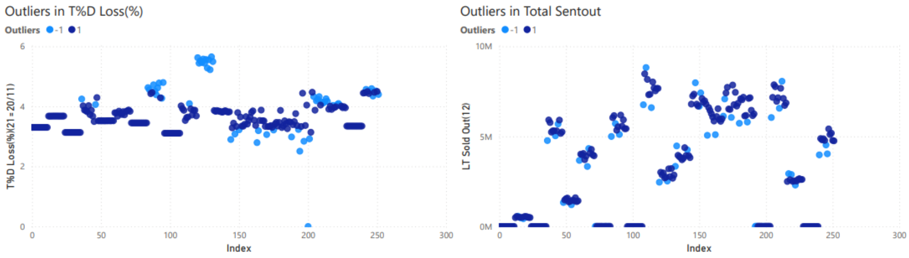

Fig. 4. Outlier Detection Graphs in Power BI

This features a plot of forecasted values compared to historical data. The forecasted values are represented through line or area charts, allowing users to evaluate the accuracy of the forecasting model. This visualization helps in understanding how well the model predicts future values and identifies any significant deviations from expected outcomes.

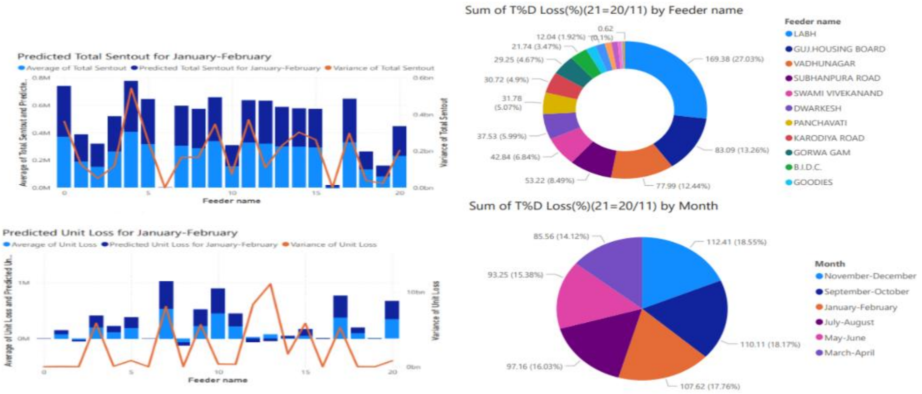

Fig. 5. Forecasting and other insights in Power BI

Level 2: Real-Time Data for Theft Detection

The second level of the Power BI dashboard focuses on real-time data for theft detection, providing tools for monitoring and analyzing live data to identify potential theft incidents.

Real Time Monitoring component displays live data feeds related to potential theft activities. Real-time visualizations, such as dynamic maps, charts, and alerts, offer immediate insights into suspicious activities as they occur. By continuously updating with the latest data, this dashboard ensures that users can monitor theft activities in real-time and respond promptly.

Theft Detection highlights unusual patterns or events that might indicate theft. These alerts are generated based on predefined criteria and data analysis. Visualizations that show historical trends and patterns related to theft. By comparing current data with historical trends, users can better understand whether ongoing incidents are part of a larger trend or isolated events.

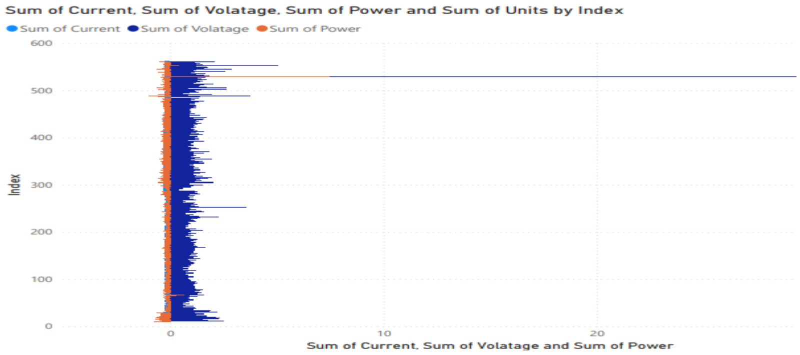

Fig. 6. Realtime smart meter data on Power BI

4.7 Smart Meter

The IoT-based smart meter system is designed for real-time monitoring of electrical parameters using an ESP32 microcontroller, ZMTP101B AC Voltage Sensor, and ACS712 Current Sensor. The ESP32 serves as the central unit of the system, managing data acquisition and communication with a remote server. The ZMTP101B sensor measures AC voltage, while the ACS712 sensor tracks the current. These sensors provide data to the ESP32, which processes this information and sends it to a server for further analysis. The server evaluates the data to determine the operational status of the electrical system and sends back signals to the ESP32. This status is visually communicated through three LEDs integrated into the system. The Green LED indicates that all parameters are within normal ranges. The Orange LED is activated when the server detects potential technical losses. The Red LED lights up in the case of potential theft. Circuit Diagram has been shown in Fig 7.

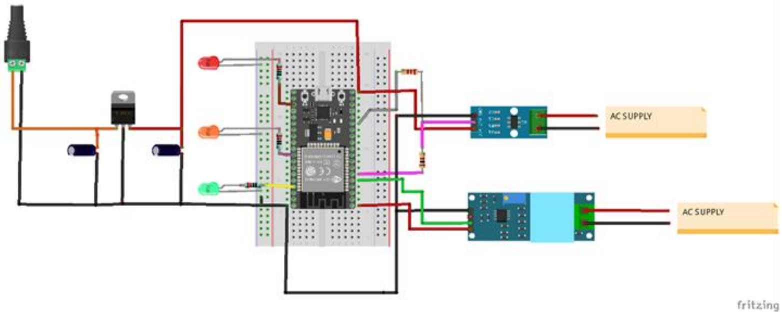

Fig. 7. Circuit Diagram of Smart Meter using ESP32

5 Results

The results from the forecasting algorithms provided valuable insights into the accuracy and performance of the models used for predicting Total Sentout, Total Billed, and Unit Loss. For the Total Sentout forecasting, the Random Forest model achieved an impressive R² score of 0.97. The Support Vector Regression (SVR) model followed with an R² score of 0.82. Slightly less effective compared to the Random Forest model. The Polynomial Regression model, attained an R² score of 0.75, Graph is shown in Fig 8.

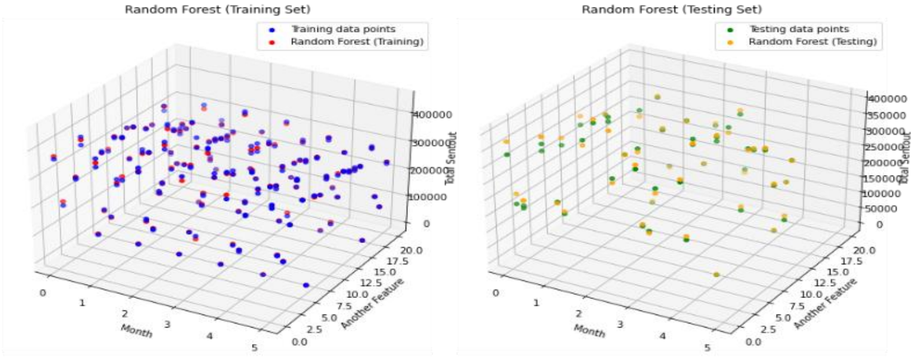

Fig. 8. Scatter Plots for Training and Testing of Random Forest Regressor on Total Sent Out

For the Total Billed forecasting, the Random Forest model again showed R² score of 0.96. The SVR model provided a comparable R² score of 0.80, while the Polynomial Regression model achieved a slightly lower R² score of 0.70 as shown in Fig 9.

In forecasting Unit Loss, the Random Forest model achieved an R² score of 0.81. The SVR model's R² score of 0.78. The Polynomial Regression model recorded an R² score of 0.72. Overall, the results demonstrate that the Random Forest algorithm consistently outperformed the other methods as shown in Fig 10

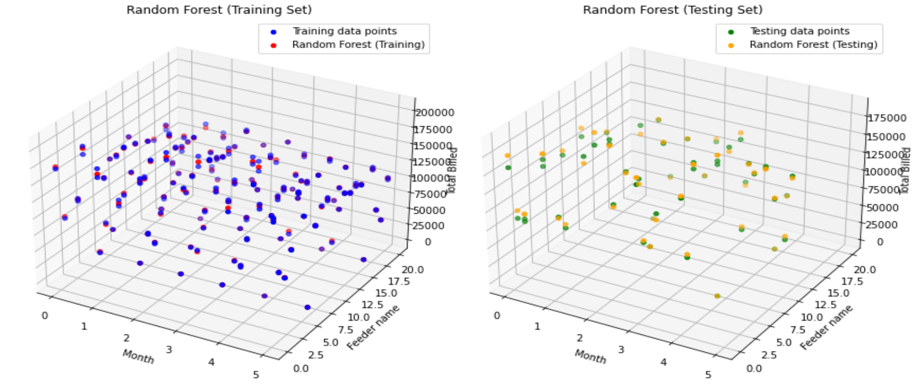

Fig. 9. Scatter Plots for Training and Testing of Random Forest Regressor on Total Billed

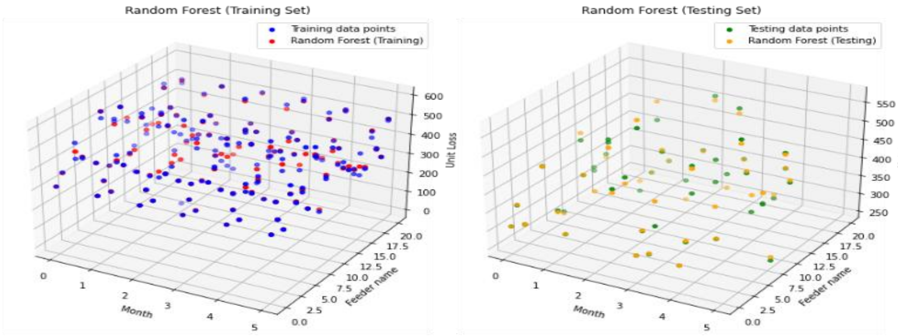

Fig. 10. Scatter Plots for Training and Testing of Random Forest Regressor on Unit Loss

There are 5 features considered by the model. X0 stands for Months and X1 stands for the feeder. X1 and X1^2 demonstrate maximum importance followed by X0*X1.

Table 2. Final R^2 Scores obtained by Random Forest Regressor

Column	Training R2 Score	Testing R2 Score
Total Sent Out	0.996136946998383	0.9728506220982995
Total Billed	0.9973531920280531	0.9687477616155546
Unit Loss	0.9014163900882001	0.8153122236175692

6 Conclusion

Entire project validated the functionality, accuracy, and performance of the smart meter system, anomaly detection and forecasting models, real-time theft detection, and Power BI dashboard. This ensures that the system is robust, reliable, and ready for deployment in practical applications.

Our outlier detection module expected to save man power, time and fuel of the Madhya Gujarat Vij Company Limited (MGVCL) in the process of theft detection. Proposed methodology will also increase the efficiency of the process thus ensuring reliability in the grid increasing its efficiency. Our prediction modules also help utility to be prepared and anticipate the metrics with average of ~92% accuracy.

7 Future Work

Future work for the research could focus on enhancing real-time data processing and predictive capabilities. Enhancing machine learning algorithms for predictive analytics could allow for early identification of potential issues by forecasting trends and anomalies before they occur. This might also include experimenting other machine learning and deep learning architectures and algorithms. In near future we are considering to integrate various data sources such as other Power Station and Main Station Data. We might also consider obtaining data from other electricity departments.

References

1. Khan, I. U., Javaid, N., Taylor, C. J., & Ma, X. (2023). Robust Data Driven Analysis for Electricity Theft Attack-Resilient Power Grid. *IEEE Transactions on Power Systems, 38*(1), 123-134.
2. El-Toukhy, A. T., Badr, M. M., Mahmoud, M., Srivastava, G., Fouda, M., & Alsabaan, M. (2023). Electricity Theft Detection Using Deep Reinforcement Learning in Smart Power Grids. *IEEE*.
3. Abdulaal, M., Ibrahem, M. I., Mahmoud, M., Khalid, J., Aljohani, A., Milyani, A. H., & Abusorrah, A. (2022). Real-time Detection of False Readings in Smart Grid AMI using Deep and Ensemble Learning. *IEEE*.
4. Elgarhy, I., Badr, M. M., Mahmoud, M., Fouda, M. M., Alsabaan, M., & Kholidy, H. A. (2023). Clustering and Ensemble Based Approach for Securing Electricity Theft Detectors Against Evasion Attacks. *IEEE Access*.
5. Zheng, K., Chen, Q., Wang, Y., Kang, C., & Xia, Q. (2021). A Novel Combined Data-Driven Approach for Electricity Theft Detection. *IEEE*.
6. Arif, A., Javaid, N., Aldegheishem, A., & Alrajeh, N. (2021). Big Data Analytics for Identifying Electricity Theft Using Machine Learning Approaches in Microgrids for Smart Communities. *Wiley*.
7. Lepolesa, L. J., Achari, S., & Cheng, L. (2021). Electricity Theft Detection in Smart Grids Based on Deep Neural Network. *IEEE*.
8. Ayub, N., Ali, U., Mustafa, K., Mohsin, S. M., & Aslam, S. (2021). Predictive Data Analytics for Electricity Fraud Detection Using Tuned CNN Ensembler in Smart Grid. *IEEE*.
9. Althobaiti, A., Jindal, A., Marnerides, A. K., & Roedig, U. (2021). Energy Theft in Smart Grids: A Survey on Data-Driven Attack Strategies and Detection Methods. *IEEE Access*.
10. Elahe, M. F., Jin, M., & Zeng, P. (2021). Review of Load Data Analytics Using Deep Learning in Smart Grids: Open Load Datasets, Methodologies, and Application Challenges. *Wiley*.
11. Althobaiti, A., Jindal, A., & Marnerides, A. K. (2021). Data-driven Energy Theft Detection in Modern Power Grids. *IEEE*.
12. Ahmed, M., Khan, A., Ahmed, M., Tahir, M., Jeon, G., Fortino, G., & Piccialli, F. (2022). Energy Theft in Smart Grids: Taxonomy, Comparative Analysis, Challenges, and Future Research Directions. *IEEE/CAA Journal of Automatica Sinica, 9*(4), 480-492.
13. Takiddin, A., Ismail, M., & Serpedin, E. (2023). Robust Data-Driven Detection of Electricity Theft Adversarial Evasion Attacks in Smart Grids. *IEEE Transactions on Smart Grid, 14*(1), 23-34.
14. (2021). A Hybrid Approach Based on Deep Learning and Support Vector Machine for the Detection of Electricity Theft in Power Grids. *Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University*.
15. Oprea, S. V., & Bâra, A. (2022). Feature Engineering Solution with Structured Query Language Analytic Functions in Detecting Electricity Frauds Using Machine Learning. *IEEE*.
16. Gunduz, M. Z., & Das, R. (2021). Smart Grid Security: An Effective Hybrid CNN-Based Approach for Detecting Energy Theft Using Consumption Patterns. *IEEE*.