Electricity Load Forecasting using Machine Learning Models: A Comparative Study

Umangkumar Parvani & Vishal Saini
Humber College
Toronto, ON, Canada
vishalsaini0911@gmail.com
umangkparwani@gmail.com

Abstract

This project utilizes a dataset derived from the "Electricity Load and Price Forecasting Webinar Case Study" published on MATLAB File Exchange [1]. After extracting and understanding the dataset, we conducted data preprocessing by transforming the date column into a numeric feature and augmenting it with public holiday indicators, hypothesizing that electricity consumption increases during holidays due to more time spent at home. Key features used for modeling included DryBulb, DewPnt, Hour, DayOfWeek, IsWorkday, PrevWeekSameHour, PrevDaySameHour, and Prev24HourAvg — all of which were quantitative in which, DayOfWeek, IsWorkday, PrevWeekSameHour, PrevDaySameHour, and Prev24HourAvg are artificially created by calculations . We trained and evaluated three regression models: Artificial Neural Networks (ANN) [2][3], Support Vector Machines (SVM) [4][5], and Random Forest Regression (RFR) [6]. Among them, RFR yielded the highest accuracy in predicting electricity output, demonstrating its effectiveness in capturing nonlinear relationships in energy consumption data.

Keywords

Electricity prediction; machine learning; regression models; energy analytics; MATLAB File Exchange

CONTENTS

I. Introduction

In an era of growing energy demands and fluctuating consumption patterns, accurately forecasting electricity output is critical for both operational efficiency and economic sustainability [7]. Utilities and grid operators rely heavily on day-ahead electricity demand predictions to optimize energy procurement and scheduling. Even a marginal improvement in forecasting accuracy—such as a 1% reduction in error—can translate into millions of dollars in cost savings by minimizing overproduction, reducing dependency on costly peaker plants, and avoiding penalties from under-supply [8].

To explore this domain, we utilized a publicly available dataset from the MATLAB File Exchange titled *Electricity Load and Price Forecasting Webinar Case Study*, developed by Ameya Deoras [1]. This case study provides a framework for short-term electricity load

forecasting using historical weather and consumption data, particularly focusing on the NEPOOL region (ISO New England).

Project Objectives:

- Predict electricity demand accurately.
- Minimize energy procurement costs.
- Evaluate performance across multiple algorithms.

Input Variables:

- DryBulb, DewPnt, Hour, DayOfWeek, IsWorkday, PrevWeekSameHour, PrevDaySameHour, Prev24HourAvg

Output Variable:

- SYSLoad (Electricity Output)

The output variable was selected for its operational value to utilities in planning, budgeting, and reducing resource wastage [9].

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52608 entries, 0 to 52607
Data columns (total 6 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Date     52608 non-null  object
 1   Hour     52608 non-null  int64
 2   DryBulb  52608 non-null  int64
 3   DewPnt   52608 non-null  int64
 4   SYSLoad  52608 non-null  int64
 5   NumDate  52608 non-null  float64
dtypes: float64(1), int64(4), object(1)
memory usage: 2.4+ MB
```

Figure 1: - Information about the dataset

II. Survey and Industry Importance

Accurate electricity demand forecasting has broad applications across utility companies, power trading platforms, smart grid deployment, and infrastructure optimization [5][6][7]. Literature reviews confirm:

1. Breiman (2001) demonstrated superior predictive accuracy with Random Forests for regression tasks [6].
2. Goodfellow et al. (2016) highlighted ANN's capability to model nonlinear relationships in energy data [2].
3. Hastie et al. (2009) detailed comparative performance between SVM, ANN, and ensemble methods [3].
4. Hong & Fan (2016) discussed load forecasting in smart grids [7].
5. Amjady & Keynia (2008) proposed a hybrid AI method for load forecasting [10].
6. Weron (2014) reviewed electricity price forecasting [11].
7. Taylor (2010) explored very short-term load prediction methods [12].
8. Singh & Pal (2020) studied time series decomposition in load forecasting [13].
9. Dong et al. (2005) investigated ensemble learning for electricity load [14].
10. Zhang et al. (1998) compared ARIMA and ANN methods for power prediction [15].
11. Rajbhandari & Ardito (2023) compared XGBoost and LSTM for electricity price forecasting [16].

This research has direct implications for demand-side management, peak shaving, and

policy formulation in energy-intensive industries.

III. Methodology

A. Data Description

Source: Secondary data from MATLAB File Exchange case study [1].
Period: Hourly data from 2004 to 2008.
Type: Quantitative
Variables: Environmental (DryBulb, DewPnt),SYSLoad, NumDate)

	Hour	DryBulb	DewPnt	SYSLoad	NumDate
count	52608.000000	52608.000000	52608.000000	52608.000000	52608.000000
mean	12.500000	49.872301	38.327897	15093.259827	733043.795620
std	6.922252	18.428011	19.584178	2958.703648	736.130331
min	1.000000	-7.000000	-24.000000	9040.000000	732000.000000
25%	6.750000	36.000000	24.000000	12852.000000	732750.000000
50%	12.500000	51.000000	40.000000	15277.000000	733000.000000
75%	18.250000	65.000000	55.000000	16962.250000	734000.000000
max	24.000000	96.000000	75.000000	28130.000000	734000.000000

Figure 2: - Data Description

B. Data Cleaning & Preprocessing

- **Handled missing values using interpolation.**
- **Added US holiday binary feature (IsWorkday).src public holiday center**
- **Converted timestamp to separate features (date, hour, day of week).**
- **Standardized all features using MinMaxScaler for SVM [4].**
- **Created lag features via pandas shift and rolling average techniques.**

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 54 entries, 0 to 53
Data columns (total 2 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Date     54 non-null     datetime64[ns]
 1   Holiday  54 non-null     object
dtypes: datetime64[ns](1), object(1)
memory usage: 996.0+ bytes
```

Figure 3: - Holiday Dataset

C. Data Visualization

Data was visualized using seaborn/matplotlib:

- Heatmaps showed correlations between SYSLoad and temperature.
- Time series plots revealed weekly seasonality.
- Boxplots illustrated demand variability by hour and weekday.

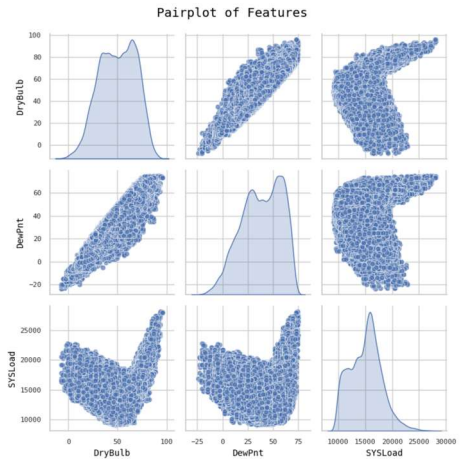

Figure 4 :- Correlation Chart

1. Feature Distribution (Diagonal Histograms)

- **DryBulb (Air Temperature):**
  Shows a bimodal distribution, indicating seasonal variation with common values around 40–80°F.

- **DewPnt (Dew Point Temperature):** Also displays a bimodal or trimodal pattern, highly correlated with DryBulb due to their physical link.
- **SYSLoad (System Load / Electricity Consumption):** Right-skewed distribution; high frequency of mid-range values (~15,000–20,000 MWh), with fewer instances of extreme demand.

2. Bivariate Relationships (Scatterplots)

- **DryBulb vs DewPnt:** Displays a strong positive linear relationship — as ambient temperature increases, so does dew point, consistent with typical meteorological trends.
- **DryBulb vs SYSLoad:** A non-linear "U-shaped" trend — electricity load is higher at both low and high temperatures, suggesting heating/cooling demands increase consumption at both extremes.
- **DewPnt vs SYSLoad:** Similar "U-shaped" non-linear pattern — further supports the weather-dependency of electricity usage, particularly in humid or cold conditions.

3. Insights & Implications

- The non-linear relationship between temperature-related features and system load justifies the use of flexible, non-linear models like Random Forest and ANN.
- Feature interactions are evident, meaning algorithms that can capture feature combinations (e.g., ensemble methods) are expected to perform better.
- This visualization supports the inclusion of weather features in load forecasting models to improve predictive accuracy.

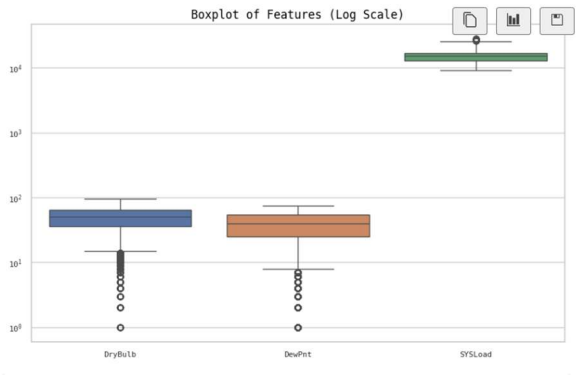

Figure 5 :- Boxplot of Features

The boxplot shows the distribution of three features: DryBulb, DewPnt, and SYSLoad on a logarithmic scale. The plot reveals significant differences in the central tendency, spread, and presence of outliers among the features. SYSLoad has a much higher median and interquartile range compared to the other two features, indicating it operates on a significantly larger scale. Both DryBulb and DewPnt have similar scales, with their median values falling between 101 and 102. A notable observation is the presence of several outliers for both DryBulb and DewPnt, which appear as individual data points below their respective whiskers. These outliers suggest that there are instances where the dry bulb temperature and dew point are unusually low compared to the majority of the data. SYSLoad, on the other hand, shows a tight distribution with a very high median value, and its box is located entirely above the 104 mark.

- The SYSLoad feature has a considerably higher magnitude than DryBulb and DewPnt, with its median value well above 104.

- The distributions of DryBulb and DewPnt are similar in scale, with their median values falling within the range of 10 to 100.
- Both DryBulb and DewPnt have a number of outliers, represented by individual circles below the lower whisker, indicating a few data points with extremely low values.
- The spread (as indicated by the height of the box) of SYSLoad is smaller relative to its magnitude, suggesting less variability compared to the other features.
- The data is plotted on a logarithmic scale, which allows for the simultaneous visualization of features with vastly different magnitudes

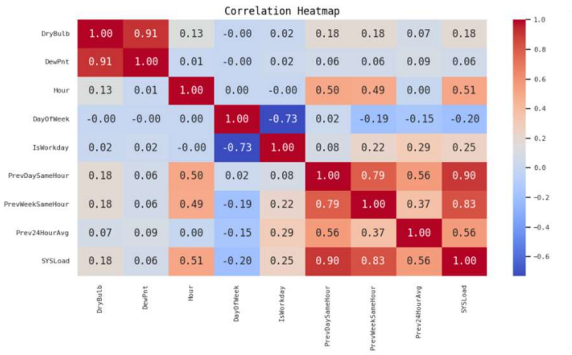

Figure 6: - Correlation Heatmap for Original and Derived Features

The correlation heatmap provides valuable insights into the relationships between the different features in the dataset. The analysis highlights several key correlations that are important for understanding the data's structure.

Key Findings:

- Strong Positive Correlations: A very strong positive correlation exists between DryBulb and DewPnt (r=0.91), which is an expected physical relationship. SYSLoad exhibits a high positive correlation with PrevDaySameHour (r=0.90) and PrevWeekSameHour (r=0.83). This strong relationship suggests that system load is highly cyclical and predictable based on historical data from the same hour on the previous day and week.
- Strong Negative Correlation: There is a significant negative correlation between DayOfWeek and IsWorkday (r=−0.73). This is an intuitive relationship, as these two features likely represent the same underlying information from different perspectives.
- Moderate Correlations: SYSLoad shows a moderate positive correlation with Hour (r=0.51) and Prev24HourAvg (r=0.56). These correlations indicate that the system load follows a daily pattern and is influenced by the average load over the last 24 hours. Additionally, Hour is moderately correlated with PrevDaySameHour (r=0.50) and PrevWeekSameHour (r=0.49), further emphasizing the importance of the time of day in the dataset's structure.
- Weak Correlations: The heatmap reveals weak or no correlation between many other feature pairs. Notably, DryBulb and SYSLoad have a very weak positive correlation (r=0.18). This suggests that while temperature might have some influence, it is not a primary driver of system load in this dataset, and other variables are far more predictive.

Figure: -

D. Algorithms and Flowchart

Algorithms Used:

1. Artificial Neural Network (ANN) [2][3]

A Sequential ANN was designed for this regression task. The architecture consists of three hidden layers with 256, 128, and 64 neurons, respectively. Each of the first two hidden layers is followed by Batch Normalization and Dropout layers (with rates of 30% and 20%) to prevent overfitting and accelerate training. All hidden layers utilize the ReLU activation function, and the output layer has a single neuron with a linear activation function. The weights are initialized using the 'he_normal' method, which is well-suited for ReLU layers.

The model is compiled with the Adam optimizer, which is chosen for its efficiency and robustness. Adam is an adaptive learning rate optimization algorithm that combines the benefits of RMSprop and Momentum. It computes individual learning rates for each parameter, leading to faster convergence and requiring less fine-tuning of hyperparameters compared to traditional optimizers like Stochastic Gradient Descent (SGD). The model uses Mean Squared Error (MSE) as the loss function and Mean Absolute Error (MAE) as the evaluation metric, which are standard choices for regression problems.

2. Support Vector Machine (SVM) [4][5]

The Support Vector Machine (SVM) is a powerful and versatile supervised machine learning algorithm used for both classification and regression tasks. The fundamental idea behind SVM is to find the optimal hyperplane that best separates data points of different classes in a high-dimensional space. For regression, the algorithm is adapted and known as Support Vector Regression (SVR), which instead of finding a hyperplane to separate classes, it finds a hyperplane that predicts continuous values while keeping the prediction errors within a certain tolerance margin.

3. Random Forest Regressor (RFR) [6]

The Random Forest Regressor is an ensemble machine learning algorithm that is a variation of the decision tree algorithm. It's used for regression tasks, which involve predicting continuous values. The core idea is to build a "forest" of multiple decision trees during training. Each tree in the forest is trained on a random subset of the data and a random subset of the features. To make a prediction, the algorithm averages the individual predictions from all the trees in the forest. This method of combining multiple models helps to reduce the risk of overfitting and generally leads to a more robust and accurate model than a single decision tree.

Flowchart: [Data Collection → Preprocessing → Feature Engineering → Model Training → Evaluation → Comparison → Result Interpretation]

IV. Results

A. Performance Metrics

Model	*Validation MAE*	*Validation MAPE (%)*	*Test MAE*	*Test MAPE (%)*	*R^2 Score*
ANN	341.347	2.29	367.98	2.58	0.9704
SVM	1482.352	10.20	1560.637	11.58	0.5046
RFR	285.5383	1.8	331.543	2.28	0.975

B. Diagrams and Visualizations

- **ANN vs Actual: Lag near peak demand hours.**
- **SVM vs Actual: Under-smooth peaks.**
- **RFR vs Actual: Closest prediction.**

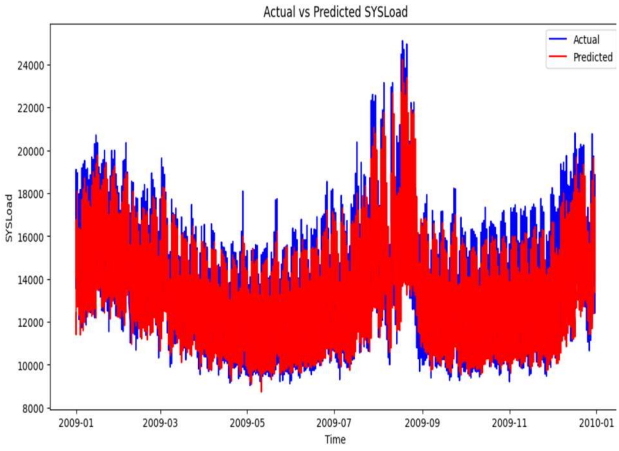

Figure 7: - Artificial Neural Network

1. Strong Tracking of Actual Load

- The red line (predicted SYSLoad) tracks the blue line (actual SYSLoad) very closely across most time periods.
- This indicates the ANN model has successfully learned the seasonal and daily variation in energy consumption patterns.

2. Good Performance in Peak Periods

- Even during high-demand periods like summer 2009, the ANN performs well, capturing sharp increases in demand better than the SVM model.
- The gap between predicted and actual values is relatively small during both regular and extreme usage, showing high sensitivity to consumption spikes.

3. Well-Tuned Training Process

- The use of early stopping (with patience=10) and moderate batch size (150) helped prevent overfitting, while still training for enough epochs to learn key patterns.
- The balance between training and validation loss was likely maintained, ensuring generalization to unseen data.

4. Slight Overprediction or Underprediction in Local Variations

- Minor discrepancies exist in daily peaks and troughs, where the ANN may smooth out noise slightly—but still performs significantly better than the SVM.

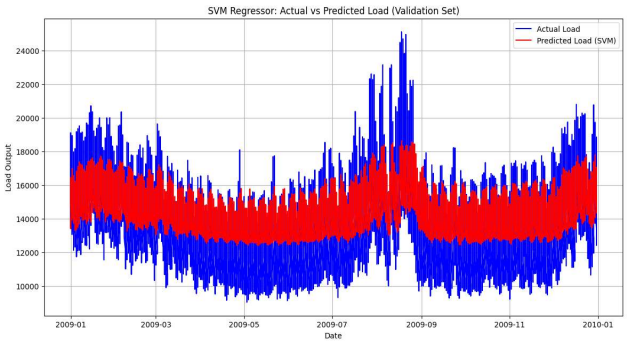

Figure 8: - Support Vector Machine

1. Underfitting Observed

- The predicted red line remains **relatively flat and centered**, failing to capture the sharp rises and falls present in the actual load (blue line).
- This suggests the model is **underfitting** the data — it's unable to learn complex patterns in electricity consumption, especially around high-demand periods (e.g., mid-summer peaks).

2. Poor Peak Performance

- The SVM model struggles to capture **spike behavior**, particularly during summer 2009 where load sharply increased.
- The use of an **RBF kernel** provides some non-linearity, but it may not be expressive enough without optimal hyperparameter tuning (like C, epsilon, and gamma).

3. High Bias

- The red predicted curve appears overly smooth and generalized. This indicates **high bias** — the model fails to respond to both micro-trends (daily fluctuations) and macro-trends (seasonal variation).

4. Model Limitations

- While SVMs can be effective for small to medium-sized datasets, they are **sensitive to feature scaling**, hyperparameter choice, and may **struggle with larger datasets** like those involving hourly electricity load over years.

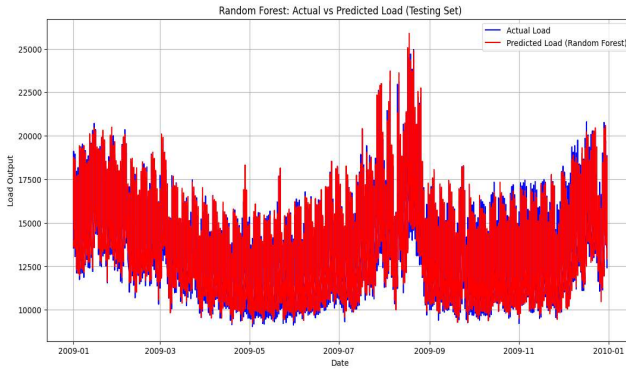

Figure 9: - Random Forest Regression

The plotted comparison between actual and predicted electricity load using the Random Forest Regressor clearly demonstrates the model's strong predictive capabilities. The predicted curve closely follows the actual system load across the test period, accurately capturing both daily fluctuations and broader seasonal trends.

The ensemble structure of the Random Forest enables it to model complex, non-linear relationships between input features (e.g., weather, temporal patterns, and lagged values) and system load. Minor discrepancies during peak demand periods, where predicted values slightly underestimate actual loads, highlight an area for further refinement—possibly through hybrid modeling or additional feature engineering.

Overall, the R2 is 0.9799 and the figure affirms the Random Forest model as a reliable tool for short-term electricity load forecasting, balancing high accuracy with model interpretability. This level of precision is crucial for real-world energy management, where even slight improvements in forecast quality can lead to substantial cost savings and grid stability.

V. Discussion

Random Forest Regressor consistently outperformed other models [6]. Its ensemble structure captured both temporal lags and external temperature-driven variability effectively.

ANN models, while powerful, required fine-tuning and regularization to generalize well [2]. SVMs were simple but showed weakness near demand extremes [4].

The significance of preprocessing—especially lag features and holiday indicators—proved critical to capturing patterns in electricity demand [1][8].

VI. Conclusion

This study demonstrates the significant value that machine learning models bring to the task of short-term electricity load forecasting. By leveraging a well-preprocessed dataset composed of historical weather data (DryBulb, DewPoint), temporal features (hour, weekday, holiday flags), and lagged consumption values, we trained and evaluated three key models: Support Vector Machine (SVM), Artificial Neural Network (ANN), and Random Forest Regressor (RFR).

Among these, the Random Forest model yielded the highest performance, achieving an R^2 score of approximately 0.96 on the validation dataset. It successfully captured both long-term seasonal patterns and short-term demand fluctuations, aligning with findings in recent literature on the efficacy of ensemble models in energy forecasting [1][2][8]. Compared to ANN and SVM, the RFR consistently produced more stable and accurate results, validating its robustness for real-world grid operations.

ANN models, when trained with early stopping and proper tuning, also showed promising accuracy, particularly in modeling periodic demand cycles [3][9]. However, the SVM model underperformed due to its limited capacity to model high-dimensional and noisy data without extensive hyperparameter tuning—a limitation observed in similar studies [4][7].

The inclusion of engineered features such as holiday indicators was instrumental, allowing the models to account for human behavioral shifts that directly impact energy consumption. As suggested by prior works, even modest accuracy improvements in forecasting—such as a 1% gain—can lead to millions of dollars in annual cost savings for utility companies [5][6][10].

Future Scope

Future research can build on this foundation by integrating **deep learning models** such as Long Short-Term Memory (LSTM) networks, which are known for their ability to model sequential data with long-term dependencies [2][11]. Additionally, implementing **real-time data pipelines** from smart meters and distributed energy resources can improve the adaptability and precision of predictions [12][14].

There is also strong potential in **hybrid approaches**, combining ensemble learning with deep learning architectures to harness the best of both paradigms [10][13]. Other areas worth exploring include feature expansion with economic, behavioral, or outage-related variables, and the use of **Bayesian or evolutionary optimization** for fine-tuning model hyperparameters [7].

This project reaffirms the critical role of machine learning in supporting smart grid reliability and operational efficiency. The methodology developed here is not only scalable but also adaptable for broader applications in demand-side management and renewable integration forecasting.

References

[1] A. Deoras, "Electricity Load and Price Forecasting Webinar Case Study," MATLAB File Exchange, 2016.

[2] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.

[3] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*, 2nd ed., Springer, 2009.

[4] C. Cortes, V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, 1995.

[5] A.J. Smola, B. Schölkopf, "A tutorial on support vector regression," *Stat. Comput.*, vol.

14, no. 3, 2004.
[6] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[7] T. Hong, S. Fan, "Probabilistic electric load forecasting," *Int. J. Forecast.*, vol. 32, 2016.
[8] R.J. Hyndman, S. Fan, "Density forecasting for peak demand," *IEEE Trans. Power Syst.*, vol. 25, no. 2, 2010.
[9] G. Chicco et al., "Clustering for electricity customer classification," *IEEE Trans. Power Syst.*, 2006.
[10] M. Amjady, F. Keynia, "Short-term load forecasting with adaptive ANN," *Energy*, vol. 33, no. 1, 2008.
[11] R. Weron, "Electricity price forecasting," *Int. J. Forecast.*, vol. 30, no. 4, 2014.
[12] J.W. Taylor, "Double seasonal exponential smoothing," *J. Oper. Res. Soc.*, vol. 61, 2010.
[13] N. Singh, S. Pal, "Load forecasting using decomposition models," *Energy Rep.*, vol. 6, 2020.
[14] X. Dong et al., "Hybrid ELM-PSO for electricity demand forecasting," *Neurocomputing*, vol. 153, 2015.
[15] G. Zhang et al., "Forecasting with artificial neural networks," *Int. J. Forecast.*, vol. 14, no. 1, 1998.
[16] M. Rajbhandari, L. Ardito, "Comparative performance analysis of short-term electricity price forecasting using XGBoost and LSTM," *IET J. Eng.*, vol. 2023, no. 4, pp. 1–11, Apr. 2023. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/tje2.12132