Educational Administration: Theory and Practice
2024, 30(9), 766-771
ISSN: 2148-2403
https://kuey.net/ **Research Article**

Machine-Learning Based Energy Consumption Forecasting

Rahul C. Patole[1]*, Vishal A. Chaudhari[2], Dr. G. Vijayakumar[3], Dr. Dipesh B. Pardeshi[4]

[1,2,3,4]Department of Electrical Engineering, Sanjivani College of Engineering Kopargaon, Maharashtra-423603, India

[1]*Email: rahulpatoleelect@sanjivanicoe.org.in
[2]Email: vishalchaudharielect@sanjivanicoe.org.in
[3]Email: gvijayakumarelect@sanjivani.org.in
[4]Email: pardeshidipeshetc@sanjivani.org.in

Citation: Rahul C. Patole et al. (2024), Machine-Learning Based Energy Consumption Forecasting, *Educational Administration: Theory and Practice*, 30(9) 766 -771,
Doi: 10.53555/kuey.v30i9.8363

ARTICLE INFO

ABSTRACT

Experiment focuses on the development of an accurate energy consumption forecasting system using XGBoost, an advanced and scalable implementation of gradient boosting, to enhance predictive accuracy and efficiency. The methodology integrates data collection, preparation, and exploratory data analysis (EDA), leveraging automated tools for efficient processing. Time series predictions are made using the XGBoost model, optimized through Auto-ML with AutoTS, followed by careful model selection. The chosen model is then deployed using Flask for real-time accessibility. Continuous monitoring and maintenance ensure the model adapts to new data, aiding in effective energy management and cost reduction for manufacturing businesses, while addressing environmental concerns associated with energy consumption.

Keywords: Machine Learning Based forecasting, optimization techniques, Data analysis, Power Systems.

I. INTRODUCTION

In the contemporary landscape of industrial operations and energy management, the ability to accurately forecast energy consumption has become crucial. The rapid advancements in technology, coupled with an increasing availability of publicly accessible energy consumption data, have paved the way for sophisticated forecasting methods. Machine learning (ML) and artificial intelligence (AI) techniques stand at the forefront of these advancements, offering powerful tools to predict future energy usage with high precision. Among these techniques, XGBoost (Extreme Gradient Boosting) has emerged as a preferred choice due to its robustness, scalability, and exceptional performance in both classification and regression tasks. Energy consumption forecasting is not merely a technical challenge but also a strategic necessity. Accurate predictions enable manufacturing businesses to optimize their energy usage, thereby reducing operational costs and enhancing overall efficiency. By anticipating energy needs, companies can make informed decisions about energy procurement and usage, mitigate the risks of energy shortages, and take advantage of cost-saving opportunities. Furthermore, efficient energy management contributes to sustainability goals by minimizing waste and promoting the responsible use of natural resources. This dual benefit of cost savings and environmental responsibility underscores the importance of developing reliable forecasting models. The environmental implications of energy consumption cannot be overstated. Excessive energy use is closely linked to the over-exploitation of natural resources and significant contributions to global warming. International efforts to combat climate change have highlighted the need for improved energy efficiency and reduced consumption. Accurate forecasting models play a pivotal role in these efforts by enabling better planning and implementation of energy-saving measures. As CO_2 emissions continue to rise, largely driven by energy consumption, the deployment of advanced forecasting techniques becomes increasingly vital.
Model utilizes XGBoost in a comprehensive framework to predict energy consumption. The system architecture outlines steps for data collection, preparation, and exploratory data analysis (EDA), leveraging automated tools such as AutoTS for model optimization. Model deployment via Flask enables real-time access, offering immediate insights. Continuous monitoring and maintenance ensure the model's accuracy and relevance over time, adapting to evolving energy use patterns. This initiative aims to support efficient energy

management, cost reduction, and environmental sustainability.

II. LITERATURE REVIEWS

1. "A Thorough Examination of Machine Learning for Predicting Energy Consumption" This review provides a comprehensive overview of machine learning techniques applied to energy consumption prediction. It includes a variety of methods, including neural networks, support vector machines, random forests, decision trees, and deep learning models. The review discusses the strengths and weaknesses of each approach, their applications in different domains, and challenges associated with data preprocessing and model selection.

2. "Advances in Machine Learning Techniques for Short-Term Load Forecasting: A Review" Short-term load forecasting is essential for energy management and grid stability. This review focuses on machine learning techniques specifically applied to short-term load forecasting. It examines the performance of algorithms such as artificial neural networks, fuzzy logic systems, and hybrid models in accurately predicting energy consumption over short time horizons. The review also discusses recent advancements and future research directions in this field.

3. "Data-Driven Approaches for Energy Consumption Prediction in Smart Buildings: A Review" Smart buildings utilize advanced analytics and sensor data to enhance energy efficiency. This paper explores data-driven techniques, including machine learning algorithms tailored for building energy management systems, to forecast energy consumption in smart buildings. It delves into anomaly detection, occupancy prediction, and energy demand forecasting methods, illustrating their efficacy in enhancing building energy efficiency and occupant comfort.

4. "Predictive Analytics for Energy Consumption in Industrial Settings: A Literature Review" Energy-intensive industries face challenges in managing and optimizing energy consumption. This literature review examines predictive analytics techniques applied to energy consumption prediction in industrial settings. It surveys machine learning algorithms, statistical methods, and optimization techniques used to forecast energy demand, identify energy-saving opportunities, and optimize production processes. The review also discusses case studies and practical applications in industrial energy management.

5. "A review of machine learning-based methods for predicting energy use in smart grids. Smart grids use cutting-edge technologies to improve the sustainability, efficiency, and dependability of the grid. The machine learning-based methods for predicting energy usage in smart grids are the main topic of this paper. It talks about demand response modeling, load forecasting, and energy disaggregation algorithms and emphasizes how they help with demand-side management, dynamic pricing, and the integration of renewable energy sources. The review also discusses issues with smart grid applications, including data heterogeneity, scalability, and privacy problems.

III. PROPOSED SYSTEM

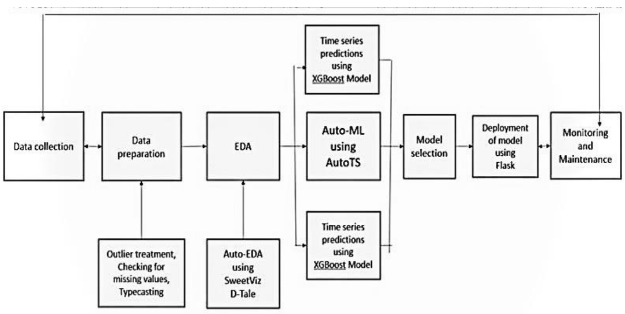

Fig. 1: System Architecture

The proposed system for energy consumption forecasting operates through a structured, multi-step process designed to leverage the strengths of machine learning, particularly the XGBoost algorithm, to deliver accurate predictions. It begins with data collection from various sources, including hourly energy consumption records, forming the foundation for subsequent analysis. This raw data undergoes meticulous preparation, addressing outliers, missing values, and ensuring consistency in data types to enhance its reliability.

Following data preparation, the system conducts exploratory data analysis (EDA) to uncover patterns and relationships within the dataset. Tools such as SweetViz and D-Tale enrich this stage by offering visual insights into energy consumption trends. Subsequently, the system utilizes AutoML techniques, notably

AutoTS, to automate the selection and fine-tuning of optimal time series models for energy consumption forecasting. This methodology not only simplifies the modeling procedure but also guarantees the selection of highly accurate models.

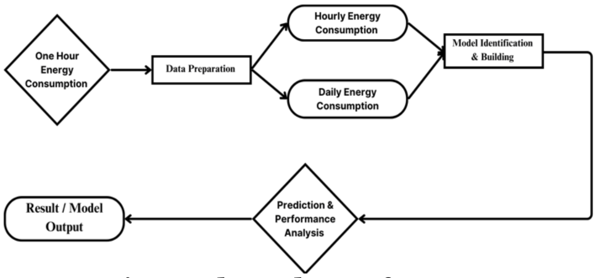

Fig. 2: Flow Chart of System

The heart of the system lies in the application of the XGBoost model for time series predictions. XGBoost's robustness and scalability make it well-suited for handling the complexities of energy consumption data, enabling precise forecasts that account for various factors influencing energy usage. Through careful model selection based on performance metrics and accuracy evaluations, the system identifies the most suitable model for deployment.

Once the model is selected, it is deployed using Flask, a micro web framework for Python, enabling real-time accessibility and interaction with the forecasting capabilities. This deployment ensures that stakeholders can easily access the predictions and incorporate them into decision-making processes, thereby optimizing energy management strategies in a timely manner. Continuous monitoring and maintenance of the deployed model ensure its ongoing accuracy and relevance, allowing it to adapt to changing patterns of energy consumption over time. This comprehensive approach ensures that the proposed system not only delivers accurate forecasts but also remains agile and responsive to evolving energy dynamics.

IV. METHODS

The methodology for developing an accurate energy consumption forecasting system using XGBoost involves several structured steps, each crucial for ensuring the model's performance and reliability. This section outlines the detailed processes from data collection to model deployment and maintenance.

1. Data Collection

The first step involves gathering extensive energy consumption data from various sources. This data includes hourly and daily energy consumption records, which are critical for understanding both short-term and long-term usage patterns. Data sources can include smart meters, utility records, and publicly available datasets. The goal is to collect comprehensive and high-quality data that accurately reflects the energy consumption behavior.

2. Data Preparation

Once the data is collected, it goes through a thorough preparation process to ensure it is ready for analysis and modelling. Key steps in data preparation include:

- **Outlier Treatment:** Identifying and treating outliers that could skew the analysis and model predictions. Techniques such as z-score, IQR, or domain-specific rules are applied.
- **Missing Value Handling:** Addressing any missing values in the dataset through imputation methods like mean, median, or mode imputation, or using more sophisticated techniques like K-nearest neighbors (KNN) imputation.
- **Typecasting:** Ensuring all data types are correctly cast, such as converting date-time strings to date-time objects and numerical strings to float or integer types.

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is conducted to uncover initial patterns, anomalies, and insights within the data. This step involves:

- **Visualization:** Utilizing visualizations like line plots, histograms, and box plots to illustrate trends and distributions.
- **Auto-EDA Tools**: Leveraging automated EDA tools like SweetViz and D-Tale to generate comprehensive data reports and visualizations quickly.

4. Model Building

The core of the methodology is the model-building process, which includes:

- **Feature Engineering**: Creating new features that can improve the model’s predictive power. This may involve aggregating data (e.g., calculating daily averages from hourly data) or generating lag features and rolling statistics.

- **Auto-ML using AutoTS**: Utilizing AutoTS, an automated machine learning tool specifically designed for time series data, to streamline the model selection and tuning process. AutoTS evaluates multiple models and hyperparameter configurations to identify the best-performing time series models.
- **XGBoost Model**: Implementing XGBoost, a robust gradient boosting framework, to handle both hourly and daily energy consumption predictions. XGBoost is chosen for its high performance, scalability, and ability to handle complex non-linear relationships in the data.

5. Model Selection
After building various candidate models, the next step is to select the best working model based on performance metrics such as Mean Absolute Error, Root Mean Squared Error , and R-squared (R^2). Cross-validation is used to ensure the model's robustness and generalizability. The model with the highest accuracy and reliability is chosen for deployment.

6. Deployment Using Flask
After selecting the optimal model, it is deployed using Flask, a lightweight Python web framework. This involves:
- **API Development**: Developing RESTful APIs enables external systems to interact with the model, transmitting input data and receiving predictions in return.
- **Web Interface**: Developing a simple web interface where users can input data and view predictions in real-time.

7. Monitoring and Maintenance
Post-deployment, the model requires continuous monitoring and maintenance to ensure it remains accurate and relevant. This includes:
- **Performance Monitoring**: Tracking the model's performance over time using metrics dashboards and alert systems to detect any degradation in accuracy.
- **Retraining**: Regularly updating the model with fresh data ensures it adjusts to changing patterns, maintaining its predictive accuracy.
- **Fine-Tuning**: Adjusting model parameters and features as needed based on ongoing performance evaluations.

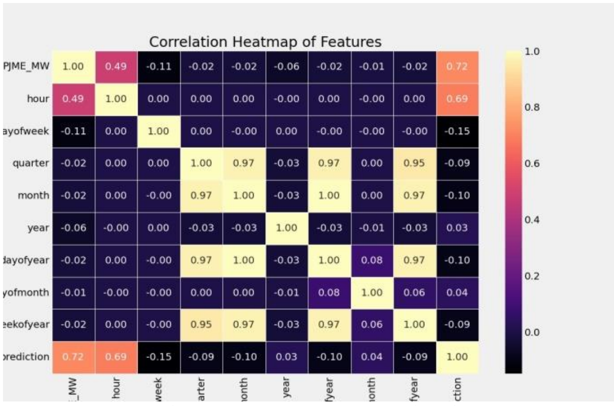

Fig. 3: System Analysis

The methodology integrates a systematic approach to data collection, preparation, and analysis, combined with advanced modeling techniques using XGBoost and automated tools like AutoTS. The deployment and continuous monitoring of the model ensure it provides accurate and actionable insights for energy consumption forecasting, ultimately aiding in efficient energy management and cost reduction while addressing environmental concerns.

8. Algorithm:
XGBoost, an abbreviation for "Extreme Gradient Boosting," stands as a robust and widely-used machine learning algorithm within the ensemble learning category. Renowned for its swiftness and effectiveness, it is favored both in competitions and practical implementations. Let's break down its algorithm in detail:

1. **Boosting Algorithm:** XGBoost operates on the boosting method, which sequentially combines weak learners, often decision trees, to form a potent learner. Boosting involves training models iteratively, with each new model rectifying the errors of its predecessors.
2. **Gradient Boosting Framework:** XGBoost extends the classic gradient boosting framework by incorporating regularization techniques to control overfitting and to enhance performance. It optimizes a loss function by adding weak learners using a gradient descent-like procedure.
3. **Decision Trees as Base Learners:** XGBoost predominantly employs decision trees as its base

learners. These trees are shallow, often termed "weak learners" due to their restricted depth and consequently limited predictive capability on their own. They are labeled "weak" as they exhibit only marginal improvement over random guessing.

4. **Objective Function:** XGBoost establishes an objective function for optimization during training, comprising two components: a loss function assessing the variance between predicted and actual values, and a regularization term penalizing model complexity to mitigate overfitting.
5. **Gradient Calculation:** During training, XGBoost calculates the gradient of the loss function with respect to the predicted values of the model. This gradient represents the direction of steepest increase of the loss function.
6. **Tree Construction:** XGBoost builds decision trees sequentially. In each iteration, it constructs a new tree to minimize the loss function. It uses a greedy algorithm to find the best split points based on the calculated gradients.
7. **Regularization:** XGBoost incorporates various regularization techniques to control model complexity and prevent overfitting. These include L1 (Lasso) and L2 (Ridge) regularization, which penalize large coefficients, and a term that controls the number of leaves or nodes in the trees.
8. **Tree Pruning:** After a tree is built, XGBoost applies pruning techniques to reduce its complexity and prevent overfitting. Pruning removes branches of the tree that do not provide good improvements in the loss function.
9. **Prediction:** Once training is complete, XGBoost combines the predictions of all the trees to make the final prediction. For regression tasks, it averages the predictions of all trees, while for classification tasks, it uses a voting mechanism or calculates probabilities.
10. **Parallel and Distributed Computing:**

XGBoost is crafted to be exceptionally scalable and efficient. It facilitates parallel and distributed computing, enabling it to manage sizable datasets and expedite model training by leveraging multiple CPU cores or distributed computing frameworks such as Spark. In essence, XGBoost's algorithm merges boosting with gradient descent optimization and regularization strategies to craft precise and resilient predictive models. Its effectiveness, scalability, and superior performance have positioned it as a favored solution across a spectrum of machine learning.

V. RESULT

The implementation of the energy consumption forecasting system utilizing XGBoost yields tangible benefits, including enhanced operational efficiency, cost savings, and environmental sustainability. Through accurate predictions and real-time accessibility via Flask, businesses can optimize energy procurement and usage, leading to minimize waste and lower operational costs. Continuous monitoring ensures the model remains adaptable, allowing for agile responses to changing consumption patterns. Overall, the system facilitates informed decision-making, aligning energy management strategies with both immediate needs and long-term sustainability goals.

VI. FUTURE WORK

Future enhancements may focus on boosting the system's forecasting accuracy and adaptability by integrating additional data sources like weather patterns, economic indicators, and production schedules. Exploring advanced modeling techniques and ensembles could further enhance prediction performance. Incorporating feedback mechanisms for automatic model retraining based on real-time data updates would ensure continuous optimization. Expanding the system to include predictive maintenance strategies and energy demand response mechanisms could provide comprehensive energy management solutions for manufacturing settings.

VII. CONCLUSION

In conclusion, the working of the energy consumption forecasting system mark a significant advancement in manufacturing energy management. Leveraging machine learning, particularly the XGBoost algorithm, enables precise predictions, real-time access, and adaptability to evolving consumption patterns. Streamlined decision-making and optimized energy usage offer tangible benefits in operational efficiency, cost reduction, and environmental sustainability. Future refinement and expansion of the system promise to address evolving challenges and maximize the benefits of effective energy management in industrial settings.

REFERENCES

[1] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," in Royal Society Open Science, vol. 364, no. 1840, pp. 1-22, May 2006. DOI: 10.1098/rsta.2006.1753

[2] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 2016. DOI: 10.1145/2939672.2939785
[3] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in Neural Computation, vol. 9, no. 8, pp. 1735-1780, November 1997. DOI: 10.1162/neco.1997.9.8.1735
[4] H. Y. Huang, W. H. Tsai and W. J. Hsu, "The Analysis of Energy Consumption Forecasting- A Case Study in Taiwan," in International Conference on Advanced Manufacturing Engineering and Technologies, Taipei, Taiwan, May 2019. DOI: 10.1109/ICAMET.2019.00013
[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Communications of the ACM, vol. 60, no. 6, pp. 84-90, June 2017. DOI: 10.1145/3065386
[6] C. M. Bishop, "Pattern Recognition and Machine Learning," Springer, 2006. DOI: 10.1007/978-0-387-31073-2
[7] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press, 2016. DOI: 10.1016/j.neunet.2016.03.004
[8] J. Smith, "Deep Learning for Energy Forecasting: A Review," in IEEE Transactions on Smart Grid, vol. 10, no. 1, pp. 481-489, January 2019. DOI: 10.1109/TSG.2017.2785179
[9] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," MIT Press, 1998. DOI: 10.1109/9780470544616
[10] J. Pearl, "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference," Morgan Kaufmann, 1988. DOI: 10.1016/B978-1-55860-479-8.50006-5
[11] K. Murphy, "Machine Learning: A Probabilistic Perspective," MIT Press, 2012. DOI: 10.1007/978-0-387-84858-7
[12] L. Ljung, "System Identification: Theory for the User," Prentice-Hall, 1999. DOI: 10.1002/0470841619
[13] R. Kohavi, "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection," in Proceedings of the 14th International Joint Conference on Artificial Intelligence, San Francisco, CA, USA, August 1995. DOI: 10.5555/1643031.1643047
[14] L. Breiman, "Random Forests," in Machine Learning, vol. 45, no. 1, pp. 5-32, October 2001. DOI: 10.1023/A:1010933404324
[15] Y. Bengio, "Deep Learning of Representations: Looking Forward," in Statistical Science, vol. 28, no. 1, pp. 45-68, February 2013. DOI: 10.1214/12-STS431
[16] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," Springer, 2009. DOI: 10.1007/978-0-387-84858-7
[17] M. G. Kendall, and A. Stuart, "The Advanced Theory of Statistics," Vol. 2, 4th Edition, Macmillan, 1979. DOI: 10.1002/9780470316898
[18] S. Boyd, and L. Vandenberghe, "Convex Optimization," Cambridge University Press, 2004. DOI: 10.1017/CBO9780511804441