Predicting Energy Demand with Interpretable Machine Learning Models

Yashwanth Karuparthi
Department of Computer Science
Birla Institute of Technology and Science Pilani Dubai Campus
Dubai, UAE
f20210136@dubai.bits-pilani.ac.in

Syed Haroon Shoaib
Department of Computer Science
Birla Institute of Technology and Science Pilani Dubai Campus
Dubai, UAE
f20210133@dubai.bits-pilani.ac.in

Harshadeep Chowdary Kommareddi
Department of Computer Science
Birla Institute of Technology and Science Pilani Dubai Campus
Dubai, UAE
f20210203@dubai.bits-pilani.ac.in

J. Angel Arul Jothi
Department of Computer Science
Birla Institute of Technology and Science Pilani Dubai Campus
Dubai, UAE
angeljothi@dubai.bits-pilani.ac.in

***Abstract*—Accurate energy demand forecasting is crucial for planning for a sustainable future. Various atmospheric conditions like temperature, humidity, wind speed, visibility and many other meteorological factors can significantly influence the demand for energy. Machine learning (ML) models utilize sophisticated algorithms to identify and interpret complicated patterns. However, the opaque nature of these models makes them difficult to understand. In this paper, we primarily present the use of QLattice, an Explainable Artificial Intelligence (XAI) model, for energy demand prediction. We used temperature and additional meteorological data for the predictions. The performance of QLattice is compared with other traditional models such as Random Forest (RF) and XGBoost. We also analyze how these models (RF and XGBoost) derive outputs by implementing an interpretability tool like Shapely Additive exPlanations (SHAP). This is compared against the inbuilt interpretability feature of QLattice. Results indicate that the QLattice models is able to achieve the highest R^2 value, MAE, MSE value of 0.993, 111 and 21316 respectively with proper feature engineering and analysis.**

Keywords—Energy Demand Forecasting; Machine Learning; Regression; Time Series Analysis; Explainable Artificial Intelligence (XAI); QLattice; XGBoost; Random Forest; Interpretability

I. INTRODUCTION

Energy demand prediction deals with identifying the amount of energy that is required by the users. In recent years, increasing environmental concerns have resulted in the need for optimization of energy usage to ensure a sustainable future. Thus, accurate energy demand prediction is important to allow policymakers, energy traders, and stakeholders in the energy sector to make informed decisions so they can optimize energy production, cost reduction, and know when to improve the system efficiency.

Machine Learning (ML) methods have been increasingly used for forecasting energy demand. This process involves analyzing vast amounts of historical data and identifying complex relationships and hidden insights within them to generate future predictions. Forecasting is a common application of ML where valuable information from the data is used either for short-term or long-term forecasting. Various statistical methods have been used for this purpose which either lacked in providing accurate results or were complex, making it difficult to understand how a model analyzes the input data, thus limiting their usefulness for decision-makers [2].

Balancing predictive accuracy with model interpretability is challenging. Understanding how ML models interpret the data can help know why a model makes certain predictions by knowing what features are considered important by the model to arrive at the output. This provides transparency to the model which is essential for policymakers to trust the predictions and better understand the factors that influence the output. Since predicting energy demand depends on many factors, interpretability helps clarify these influences and supports more informed decision-making.

This work uses QLattice, an XAI model, to obtain highly accurate energy demand prediction. The objective is to evaluate the performance of QLattice and compare it to conventional ML models, while also using it to interpret results. Furthermore, we attempt to improve the accuracy of both QLattice and the other ML models by incorporating lag and rolling window statistical features to help the models accommodate the dynamic nature of the data that changes over time [3]. Additionally, we also check the importance of various weather features other than temperature by comparing their impact on the accuracy. This study is the first attempt to use QLattice in the long-term prediction of energy demand with multiple climate variables.

The methodology for this study involves the following steps: The energy demand and weather datasets are first combined and converted from hourly to daily frequency using mean aggregation. New features are introduced by generating daily statistical features on weather variables to capture trends and outliers. Lag and rolling window features on the demand data are generated along with introducing interactive features. Finally, XGBoost, Random Forest, and QLattice are employed in this study to test the hypothesis mentioned earlier.

This research work is structured into 7 sections: Section 2 provides an overview of the related works used for the task of forecasting energy demand. Section 3 describes the dataset exploring the features used for this study. After describing the applied method in Section 4, we discuss the implementation of the proposed model in Section 5. The results obtained from the implementation are presented in Section 6 which leads us to Section 7 where we conclude the hypothesis of this research.

II. LITERATURE SURVEY

This section reviews the previous works in the area of energy demand prediction. A. González-Briones et al. [4] used 2 years of data from a shoe store in Salamanca, Spain for

979-8-3503-5348-8/24/$31.00 ©2024 IEEE

predicting energy consumption using models like Random Forest (RF), Decision Tree (DT), K-Nearest Neighbors (KNN), Support Vector Regression (SVR), and Linear Regression (LR). After the training and testing phase, LR and SVR turned out to be the best models with both of them having an accuracy of 0.857.

D. K. Moulia et al. [5] used the Domestic Electrical Load dataset to predict the energy consumption of residential buildings in South Africa. The paper used four different ML models namely RF, DT, XGBoost, and AdaBoost to predict hourly consumption. The results showed that both DT and RF had an R^2 value of 0.99, while the boosting models had R^2 values of 0.98 and 0.74 respectively. It was concluded that the implementation of ensemble learning models might improve the accuracy of predictions.

A data-driven hybrid model (ISCOA-LSTM) was presented by N. Somu et al. [6] that employed a long-short term memory network and an improved sine cosine optimization algorithm. The resultant model outperformed other state-of-the-art ML algorithms on real-time energy consumption data obtained from the Indian Institute of Technology, Bombay. The proposed model showed an RMSE value of 0.0559. Despite its success, this work lacks the analysis of the impact of weather data on model performance.

Previous work on using QLattice for energy-related use cases was done by S. Wenninger et al. [7] where QLattice was implemented along with ML algorithms like Artificial Neural Networks (ANNs), SVR, XGBoost, and Multiple Linear Regression (MLR) to compare the prediction performance. The dataset contained information based on two-family buildings from Germany with 74 variables describing the building characteristics. The paper compared the performance of the models and observed that XGBoost gave the best performance.

Y. Shimizu et al. [8] used weather data to predict household energy consumption by collecting data for 18 selected houses in Okinawa, Japan over one year. Five weather parameters like humidity, wind speed, solar radiation, rainfall, and temperature were used with a sparse regression method to train the model and performed 10-fold cross-validation for evaluation. It was concluded that weather data can contribute to a good accuracy score for predicting energy consumption.

III. DATASET

Two datasets related to Ontario are used in this work which are historical energy demand data and Ontario weather data. The former dataset consists of hourly data and is available in CSV file format on the Independent Electricity System Operator's website where nineteen files are manually downloaded from the source. The latter dataset consists of hourly weather details of Ontario province of Canada. The weather data is derived from 169 unique weather stations in Ontario, provided by the Canadian Ministry of Environment [9], and covers the period from January 1994 to December 2018. The weather dataset includes features such as temperature, dew point temperature, relative humidity, wind speed, visibility, wind chill, pressure, humidex, and weather. The energy demand dataset has 222,096 samples and includes information on date, time, and corresponding hourly demand. The weather dataset, with 582,248 samples, includes key weather features such as temperature, dew point temperature, relative humidity, pressure, visibility, and humidity index. The weather attribute describes the general state of the weather in string format. Upon compiling both energy demand and weather datasets and cleaning the missing data, the resultant dataset consists of 219,144 samples.

IV. METHODOLOGY

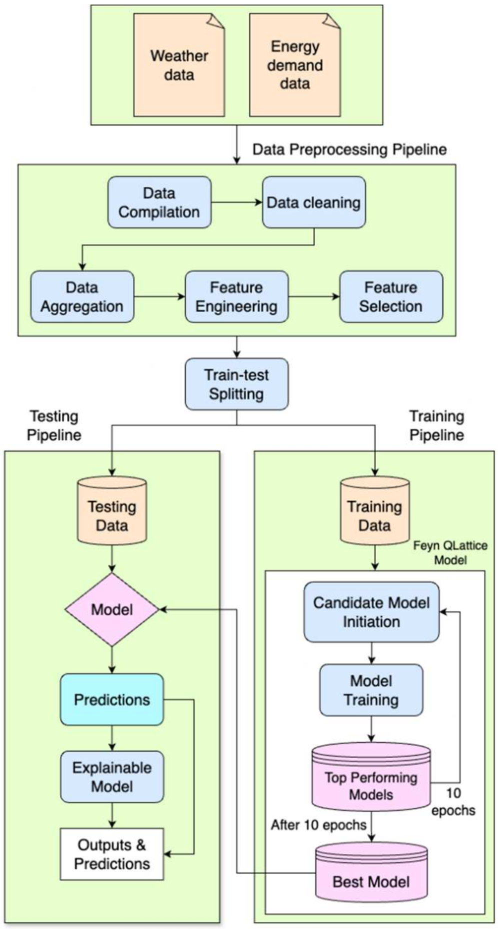

Fig. 1. Model architecture of the proposed QLattice-based energy demand prediction

Fig. 1. provides the architecture of the proposed work. Weather and energy datasets are combined before undergoing preprocessing. The data is then cleaned, additional features are added, and irrelevant features are removed. The resulting dataset is split into testing and training subsets. Training data is used by the QLattice algorithm to train several predefined models before the best one is selected. This model is then used on the testing data to obtain predictions and perform evaluation.

A. *Data Preprocessing*

1) *Data Compilation*

The hourly energy demand data and weather data which are obtained in multiple files are consolidated. All the energy demand files are combined into a single CSV file. Similarly, the weather data files are merged into one CSV file, resulting in a unified weather dataset.

2) *Data Cleaning*

The demand dataset has a single full month of data missing for December 2001. Fortunately, due to monthly consistent patterns in the demand data, the data from December 2000 is chosen as replacement data considering the average monthly temperature being close to that of the missing data. There is also some data missing within the temperature feature, where out of 582,248 samples only 185 samples are missing, which accounted for only 0.03% of the total data and are randomly scattered throughout the dataset, hence they are imputed using linear interpolation.

3) Data Aggregation

The dataset which is initially available in hourly frequency is converted to daily frequency to simplify the analysis and reduce model computational complexity. This is done by using the mean aggregation method. The energy demand and weather datasets are compiled into a single power and weather multi-year dataset based on the date-time index with each row having hourly data on Ontario energy demand and weather details.

4) Feature Engineering

In the feature engineering phase, various techniques are used to enhance the predictive power of the models. One critical approach is introducing various statistical features. Daily mean and median features are introduced for weather variables to reduce the effect of outliers, smooth out short-term fluctuations and highlight longer-term trends.

Regarding energy demand in correlation with weather, it is often influenced by previous weather conditions. To incorporate past information from the energy demand data and key weather features into the model, lag features are generated on demand data with the lag timeframe ranging from one day to seven days period to capture the influence of past values. This is based on the results obtained from the work of C. Zhang et al. [10], and H. S. Lim and G. Kim [11]. In addition to lag features, for the model to identify cyclic patterns, rolling mean and standard deviation of energy demand with window sizes of 3 and 7 respectively are added [12]. To capture the relationship between different weather variables on energy demand, interactive features were introduced from temperature combined with humidity and wind speed data separately.

5) Feature Selection

The weather attribute in the weather dataset has about 157 unique string values describing the general weather state across the Ontario region and has 27,797 missing values that accounted for 12.68% of the total data. Considering the high cardinality of the attribute with 157 unique string values and its missing rate, combined with the availability of similar information from other attributes, it is decided to drop the attribute to mitigate the risk of dimensionality issues and improve model efficiency.

Hence the final features that are included in the research analysis are temperature, dew point temperature, relative humidity, wind speed, visibility, pressure, wind chill, humidex, daily demand, daily minimum, maximum, mean, and median values of temperature, dew point temperature, relative humidity, visibility, pressure, humidex, rolling mean values with 3 day window and rolling standard deviation values with 7 days window of energy demand, 1 to 7 days lag variables of energy demand, and interaction features between temperature and humidity as well as temperature and wind speed.

B. Energy demand prediction using QLattice

QLattice [13], developed by Abzu and inspired by Richard Feynman's path integral formulation, builds simple mathematical models to derive relationships between the variables. This model eliminates the black box concept in various ML models by providing explanations on the predictions. QLattice searches thousands of potential models to find the ideal one and seek the best features to solve a computation problem. The models are in the form of simple mathematical expressions which are presented as unidirectional and acyclic graph by QLattice, and therefore allow researchers to study the relationships between the variables. QLattice generates models by getting a list of possible expressions to obtain the output from the input parameters and then sorting them by various evaluation metrics. The list of possible expressions is obtained from a set of all possible expressions [7].

C. Explainability

An important downside of Artificial Intelligence (AI) models is that there is no clarity over how exactly a model arrives at a prediction. While we know the general processes that the algorithm will follow to build the model, it is not clear what the "reasoning" behind a specific prediction is. AI models are hence "black boxes". To promote trust within the public over the results of AI predictions, and to detect and fix bias, explainability of the models is necessary. For this purpose, XAI tool like SHapley Additive exPlanations (SHAP) is used.

1) SHAP

SHAP, is an XAI technique developed in 2017. It explains the predictions of black box AI models by assigning all the features of the dataset with a "Shapely value" that indicates how impactful that feature is in determining the final prediction of the model. SHAP can also be used to generate descriptive graphs for easier visualization and analysis.

Shapley values are calculated using (1). Let f be the black box model, ϕ_i denote the shapley value for feature *i*, *M* denote the set of all features, z' denote a subset of features not including feature *i*, and *f(z')* denote the output of the model for features including *i*.

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!\,(M - |z'| - 1)!}{M!}\left[f_x(z') - f_x(z' \setminus i)\right] \quad (1)$$

The output when a subset of features without *i* is used as input is compared to the output with feature *i*. This value is then adjusted based on the frequency of appearance of i in possible feature subsets. The final shapley value is the summation of the results of this process for all possible feature subsets [14].

V. IMPLEMENTATION

All models in this work are implemented in the Python 3.12.3 programming language and using the Jupyter Notebook. Several libraries used for implementing the models with their specific uses are detailed as follows: The data mining algorithms along with the methods used for evaluating the models are imported from the Scikit-learn library; QLattice is implemented using the Feyn library; Matplotlib library provides functions for making graphs; the NumPy library is used for importing methods to work with arrays; Pandas library is used to work with the dataset; SHAP is implemented by importing the SHAP Python library. An 8GB RAM MacBook Air with an M2 processing chip is used to run the code. The dataset is split in the ratio of 67% (training set) and 33% (testing set).

In this work, ML algorithms like XGBoost and RF are used for comparison. XGBoost is an ML algorithm developed in 2016 by T. Chen and C. Guestrin [15]. It uses an ensemble learning technique called gradient boosting, in which models are built sequentially, with each new iteration making corrections in the previous one, thereby building a highly accurate model. XGBoost builds DTs that act as base learners. Each tree fixes the errors of the previous one. The loss function is minimized by using the loss function gradient to guide the formation of new trees. RF [16] is a ML technique developed by L. Breiman in 2001 which uses the bagging technique. It works by creating many DTs, each one made using a randomly selected subset of the training data. After the construction of the DTs, the ensemble learning technique combines the individual predictions from the DTs to get final predictions.

For evaluating all the models, Mean Absolute Error (*MAE*), Mean Squared Error (*MSE*), Root Mean Squared Error (*RMSE*), and R-squared (R^2) are used. Mean Absolute Error (*MAE*), as shown in (2), is the average of the absolute differences between the predicted values ($\hat{y}$) and the actual values (y_i).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}| \tag{2}$$

Equation (3) signifies Mean Squared Error (*MSE*), which is the average of the squared differences between predicted values ($\hat{y}$) and the actual values (y_i).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2 \tag{3}$$

Root Mean Squared Error (*RMSE*), (4), can be viewed as the square root of the Mean Squared Error (*MSE*).

$$RMSE = \sqrt{MSE} \tag{4}$$

The coefficient of determination (R^2) can be termed as the proportion of variance in the dependent variable. Its values range from 0 to 1, where the value closer to 1 indicates the better fit of the model to the data. (5) is the mathematical representation of R^2.

$$R^2 = 1 - \frac{\sum(y - \hat{y})^2}{\sum(y - \bar{y})^2} \tag{5}$$

VI. Results

This section details the results of all the experiments conducted.

A. Energy demand prediction using only temperature data

To understand the effect of temperature variables on the energy demand prediction task, we conduct this experiment by using only the temperature variables from the dataset. Table I presents the results of this experiment. The regression results from Table I show that RF achieves the best performance and shows a good fit with the actual data. This is evident from the R^2 value of 0.938 and the MAE value of 320.57. This was comparable to the results obtained using XGBoost ($R^2 = 0.937$ and MAE = 330.172). Both indicate that temperature data alone, without additional feature engineering is sufficient to achieve acceptable results. QLattice on the other hand shows poor returns, with an R^2 value of 0.775 and MAE of 650.

TABLE I. Performance of the Models Using Only Temperature Data

Algorithm	**Evaluation metrics**			
	MAE	***MSE***	***RMSE***	***R²***
XGBoost	330.17	189828.41	435.69	0.937
Random Forest	320.57	186772.84	432.17	0.938
QLattice	650	674041	821	0.775

B. Energy demand prediction using only weather data

In order to understand the effect of weather variables on the energy demand prediction task, we conduct this experiment by using only the weather variables from the dataset. Table II presents the results of this experiment. The evaluation metrics change very little when only weather data is used. The XGBoost and Random Forest shows R^2 values of 0.934 and 0.931, along with MAE values of 339.479 and 335.372 respectively. QLattice also shows only a very slight change with an R^2 value of 0.777 and an MAE value of 653.

TABLE II. Performance of the Models Using Only Weather Data

Algorithm	**Evaluation metrics**			
	MAE	***MSE***	***RMSE***	***R²***
XGBoost	339.47	199711.621	446.891	0.934
Random Forest	335.37	208418.127	456.528	0.931
QLattice	653	665856	816	0.777

C. Energy demand prediction by combining temperature and weather data

To improve results, an experiment is performed by combining weather and temperature data. The results of this experiment are shown in Table III. The regression results using XGBoost shows a good fit with the actual data, with the predictions explaining over 93.8% of the variance within the complete dataset (R^2 = 0.938). The mean absolute error (MAE) for all predictions is 329.03. This is comparable to the results obtained using RF (R^2 = 0.935 and MAE = 327.90). Both of these indicate that weather and temperature data alone, without additional feature engineering is sufficient to achieve acceptable results. QLattice on the other hand shows poor results, with an R^2 value of 0.768 and MAE of 662.

D. Energy demand prediction after feature engineering

In order to further improve the results, new features such as rolling mean, rolling standard deviation with different window sizes and lag features are added after feature engineering process. Table IV shows the performance of all the models on using these new features. From Table IV, it can be seen that the performance of all the models increases significantly due to use of these new features. Amongst all the models experimented with, QLattice shows the highest performance with R^2 value of 0.993, and MAE of 111, while RF is the ML model that exhibited the highest performance with an R^2 value of 0.984, and MAE of 145.81.

TABLE III. PERFORMANCE OF THE MODELS UPON COMBINING TEMPERATURE AND WEATHER DATA

Algorithm	Evaluation metrics			
	MAE	***MSE***	***RMSE***	***R²***
XGBoost	329.03	188520.57	434.18	0.938
Random Forest	327.90	197085.06	443.94	0.935
QLattice	662	668900	830	0.768

TABLE IV. PERFORMANCE OF THE MODELS AFTER FEATURE ENGINEERING

Algorithm	Evaluation metrics			
	MAE	***MSE***	***RMSE***	***R²***
XGBoost	199.56	72754.51	269.73	0.976
Random Forest	147.81	45893.87	214.22	0.984
QLattice	111	21316	146	0.993

Fig. 2. shows a plot of the energy consumption values predicted by the QLattice model and the actual energy consumption values for June 2017. It can be seen from the figure that the QLattice model can predict the values very close to the actual values.

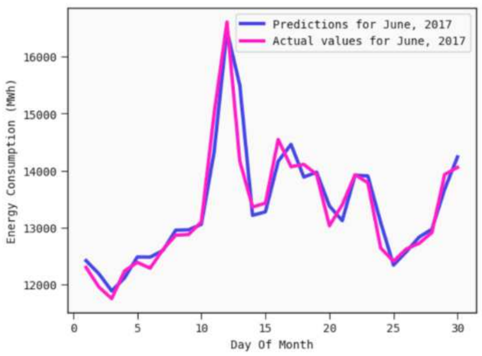

Fig. 2. Plot of energy demand values predicted by the QLattice model and the actual values for June 2017.

E. *Interpretability analysis*

As the XGBoost model is the best-performing model, we chose this model for the interpretability analysis using the SHAP. Fig. 3. shows the SHAP plot for the top 6 features of the XGBoost model. It can be seen from the figure that daily mean temperature values has the most significant impact on the predictions, followed by the year, day of the week, sunset times, week of the year, and daily median temperature.

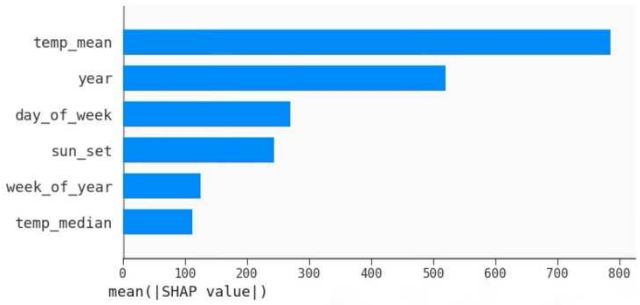

Fig. 3. SHAP plot for the XGBoost model for the top 6 features.

From the SHAP summary plot, shown in Fig. 4., lower mean temperatures results in more positive SHAP values, while higher daily mean temperatures results in more negative SHAP values. It can be inferred from this that, in general, higher temperatures resulted in the lowering of electricity demand, while lower temperatures increases the demand.

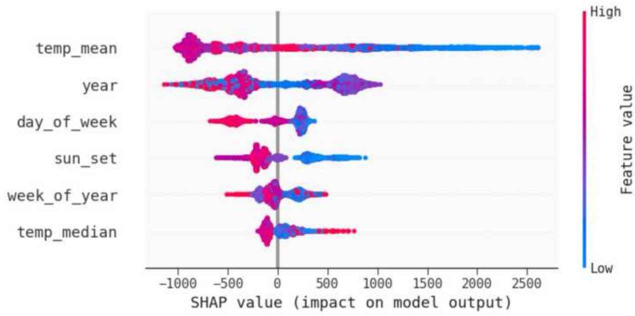

Fig. 4. SHAP summary plot for the XGBoost model.

Using the dependence plot in Fig. 5., it can also be seen that more extreme the temperature (below 0°C and above 30°C) the more positive the impact of daily mean temperature was, while the closer the mean temperature was to the 10°C-20°C range, the more negative the impact was.

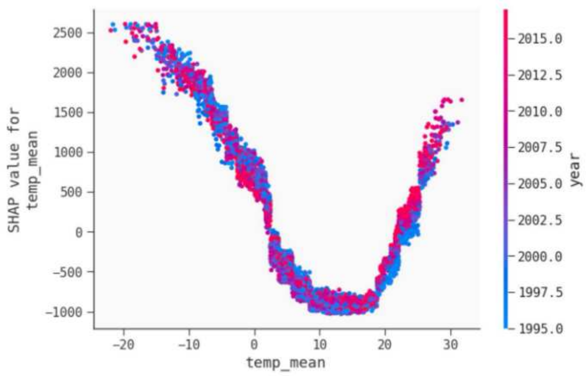

Fig. 5. SHAP dependence plot for the XGBoost model.

After further feature engineering the RF model exhibits better performance than the XGBoost model. It can be seen from Fig. 6. that demand rolling mean and demand lag are by far the most impactful features for the RF model, with the rest of the features having almost negligible SHAP values.

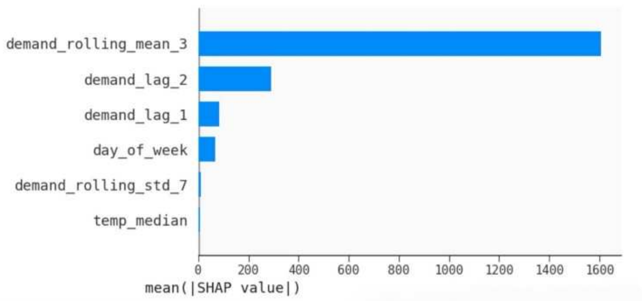

Fig. 6. SHAP plot for the RF model for the top 6 features.

The QLattice model graph in Fig. 7., shows that 5 input variables were used to predict the daily demand. Of those, demand rolling mean had the highest Mutual Information value of 1.49, followed by demand lag with 1.22, daily minimum temperature with 1.21, visibility with 1.07 and lastly daily maximum pressure with 0.97. Daily demand overall had an MI value of 1.64. These results imply that demand rolling mean was the most impactful feature for prediction, followed by demand lag and daily minimum temperature. The model graph can further be used to see the expression that is used to predict the output values.

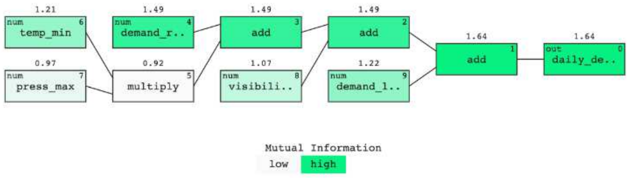

Fig. 7. QLattice model graph.

VII. Conclusions And Future Scope

Energy demand prediction is a complex task with the presence of several complex patterns in the data making it difficult for conventional statistical techniques to achieve accurate prediction. QLattice is a recent model that exhibits the ablity to achieve reasonable performance along with interpretability. It has been observed in this work that after feature engineering, QLattice performs better than both XGBoost and Random Forest for the energy prediction task. The inbuilt interpretability feature of QLattice also performed satisfactorily, and gave an understanding on the most important features for prediction. However, it is less informative than the SHAP XAI which was used for comparison. It can also be seen that the addition of rolling window features caused a significant improvement in predictions for both QLattice, and the RF and XGBoost models. Further research can be conducted by incorporating economic and social factors such as GDP, energy prices and poverty levels for improvement of predictive forecasts.

References

[1] A. B. Arrieta *et al.*, “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI,” *Information Fusion*, vol. 58, pp. 82–115, Jun. 2020, doi: 10.1016/j.inffus.2019.12.012.

[2] L.-V. Herm, K. Heinrich, J. Wanner, and C. Janiesch, “Stop ordering machine learning algorithms by their explainability! A user-centered investigation of performance and explainability,” International Journal of Information Management, vol. 69, p. 102538, Apr. 2023, doi: 10.1016/j.ijinfomgt.2022.102538.

[3] I. A. Khan, A. Akber, and Y. Xu, “Sliding Window Regression based Short-Term Load Forecasting of a Multi-Area Power System,” in 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), May 2019. Accessed: Aug. 20, 2024. [Online]. Available: http://dx.doi.org/10.1109/ccece.2019.8861915

[4] A. Gonzalez-Briones, G. Hernandez, J. M. Corchado, S. Omatu, and M. S. Mohamad, “Machine Learning Models for Electricity Consumption Forecasting: A Review,” in 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), May 2019. Accessed: Aug. 20, 2024. [Online]. Available: http://dx.doi.org/10.1109/cais.2019.8769508

[5] D. K. Moulla, D. Attipoe, E. Mnkandla, and A. Abran, “Predictive Model of Energy Consumption Using Machine Learning: A Case Study of Residential Buildings in South Africa,” Sustainability, vol. 16, no. 11, p. 4365, May 2024, doi: 10.3390/su16114365.

[6] N. Somu, G. R. M R, and K. Ramamritham, “A hybrid model for building energy consumption forecasting using long short term memory networks,” Applied Energy, vol. 261, p. 114131, Mar. 2020, doi: 10.1016/j.apenergy.2019.114131.

[7] S. Wenninger, C. Kaymakci, and C. Wiethe, “Explainable long-term building energy consumption prediction using QLattice,” Applied Energy, vol. 308, p. 118300, Feb. 2022, doi: 10.1016/j.apenergy.2021.118300.

[8] Y. Shimizu, T. Sakagami, and H. Kitano, “Prediction of weather dependent energy consumption of residential housings,” in *2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA)*, Nov. 2017. Accessed: Aug. 20, 2024. [Online]. Available: http://dx.doi.org/10.1109/icrera.2017.8191202

[9] C. Kolyvas, S. Missiakoulis, and F. Gofa, “Mean Daily Temperature Estimations and the Impact on Climatological Applications,” in 16th International Conference on Meteorology, Climatology and Atmospheric Physics—COMECAP 2023, Aug. 2023. Accessed: Aug. 20, 2024. [Online]. Available: http://dx.doi.org/10.3390/environsciproc2023026131

[10] C. Zhang, J. Li, Y. Zhao, T. Li, Q. Chen, and X. Zhang, “A hybrid deep learning-based method for short-term building energy load prediction combined with an interpretation process,” Energy and Buildings, vol. 225, p. 110301, Oct. 2020, doi: 10.1016/j.enbuild.2020.110301.

[11] H. S. Lim and G. Kim, “Prediction model of Cooling Load considering time-lag for preemptive action in buildings,” Energy and Buildings, vol. 151, pp. 53–65, Sep. 2017, doi: 10.1016/j.enbuild.2017.06.019.

[12] E. Zivot and J. Wang, “Rolling Analysis of Time Series,” in Modeling Financial Time Series with S-PLUS, Springer Science & Business Media, 2003.

[13] K. R. Broløs et al., An Approach to Symbolic Regression Using Feyn, Apr. 2021, [online] Available: http://arxiv.org/abs/2104.05417.

[14] S. M. Lundberg and S.-I. Lee, ‘A unified approach to interpreting model predictions’, Advances in neural information processing systems, vol. 30, 2017.

[15] 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, Aug. 2016. Accessed: Aug. 20, 2024. [Online]. Available: http://dx.doi.org/10.1145/2939672.2939785

[16] L. Breiman, ‘Random forests’, *Machine learning*, vol. 45, pp. 5–32, 2001