The Scientific Temper (2024) Vol. 15 (spl): 230-237
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.27
E-ISSN: 2231-6396, ISSN: 0976-8653
https://scientifictemper.com/
RESEARCH ARTICLE

Smart grid precision: Evaluating machine learning models for forecasting of energy consumption from a smart grid

B. S. E. Zoraida*, **J. Jasmine Christina Magdalene**

Abstract

The widespread adoption of smart home technologies has led to a significant increase in the generation of high-frequency energy consumption data from smart grids. Accurate forecasting of energy consumption in smart homes is crucial for optimizing resource utilization and promoting energy efficiency. This research work investigates the precision of energy consumption forecasting within a smart grid environment, employing machine learning algorithms such as convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM), temporal fusion transformer (TFT) and Prophet. The CNN model extracts spatial features, while RNN and LSTM capture temporal dependencies in time series data. Prophet, recognized for handling seasonality and holidays, is included for comparative analysis. Utilizing a dataset from Pecan Street, Texas, performance metrics like mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE) assess each model's accuracy. This work aids in improving energy management systems, contributing to sustainable and efficient energy use in residential environments.

Keywords: Smart grid, Recurrent neural network, Long short-term memory, Temporal fusion transformer, Prophet.

Introduction

In the contemporary landscape of urban living, smart grids have emerged as a requirement in the quest for energy efficiency and sustainability. To appreciate the significance of energy consumption forecasting within a smart grid, it is imperative to trace the evolution of smart grids themselves. Traditionally, electricity grids operated in a unidirectional manner, with energy flowing from centralized power plants to end-users. However, the advent of renewable energy sources, coupled with the proliferation of distributed energy resources, has necessitated a more dynamic and adaptable grid infrastructure. Smart grids represent a transformative response to these challenges. These intelligent networks integrate advanced detection, communication, and control technologies to improve the consistency, efficacy and sustainability of electricity distribution. The two-way flow of information between consumers and the grid enables real-time monitoring and optimization, fostering a more responsive and resilient energy ecosystem. The integration of advanced technologies within the electricity grid has given rise to a paradigm shift in the way energy is generated, distributed and consumed.

At the heart of this progression lies the concept of smart homes, where the continuous integration of devices, sensors, and communication protocols enables residents to monitor and control various aspects of their living spaces remotely. From smart thermostats that adjust heating and cooling based on usage patterns to connected appliances that optimize energy consumption, these homes generate a wealth of high-frequency data. Smart homes, therefore, present a unique challenge and opportunity in the realm of energy consumption forecasting. The heterogeneity of devices, coupled with the dynamic nature of human behavior, necessitates forecasting models that can adapt and respond to evolving patterns.

This comprehensive exploration delves into the intricate dynamics of energy consumption within a smart grid environment, controlling cutting-edge machine

Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.

***Corresponding Author:** B. S. E. Zoraida, Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, Tamil Nadu, India, E-Mail: smitha.b@bdu.ac.in

How to cite this article: Zoraida, B.S.E., Magdalene, J.J.C. (2024). Smart grid precision: Evaluating machine learning models for forecasting of energy consumption from a smart grid. The Scientific Temper, **15**(spl):230-237.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.27

Source of support: Nil

Conflict of interest: None.

Received: 17/09/2024 **Accepted:** 29/09/2024 **Published**: 16/10/2024

learning algorithms to unravel the complexities inherent in this expanding domain. The algorithms that are used encompass a diverse spectrum, including convolutional neural network (CNN), recurrent neural network (RNN), long short-term memory (LSTM), Prophet and temporal fusion transformer (TFT). Each of these algorithms brings a unique set of capabilities, offering a mosaic of approaches to tackle the multifaceted challenge of energy consumption forecasting. The accuracy of these models directly impacts the optimization of resource utilization, cost savings and the overall sustainability of energy consumption in smart homes.

The dataset for this research work is derived from Pecan Street, Texas. The dataset encapsulates historical records of energy consumption, offering a real-world context that enriches the evaluation of machine learning algorithms. The chosen algorithms undergo rigorous training and evaluation against this dataset, employing well-established performance metrics. These metrics serve as benchmarks, quantifying the accuracy of each forecasting model and facilitating a comparative analysis.

The experimental results unfold the understanding of the strengths and limitations of each algorithm in predicting energy consumption patterns over varying seasons. The insights gathered from this comprehensive analysis contribute not only to the field of machine learning but also to the practical implementation of forecasting models within the dynamic context of smart homes. Beyond the confines of algorithmic selection, the findings of this work have broader implications for the enhancement of energy management systems in smart homes. The practical insights acquired here pave the way for sustainable and efficient energy usage in residential environments, aligning with the broader goals of energy conservation and environmental stewardship.

In essence, this research not only navigates the intricacies of energy consumption forecasting within the context of a smart grid but also charts a course toward a more intelligent and sustainable future for smart homes. As technology continues to evolve and smart homes become increasingly integral to the fabric of urban living, the quest for accurate and adaptable energy consumption forecasting becomes not just a necessity for the consumers but also to the utilities.

Related Works

The researchers have looked at the analysis of stock market data using machine learning techniques, particularly recurrent neural networks (RNNs) and long short-term memory (LSTM) models. For the purpose of predicting groundwater, Aditi Singh.,(2023).

The researchers have compared local LSTM models trained on local well data with global LSTM models calibrated across all monitoring well data. Using LSTM-based forecasting models, Sumriti.,(2023).

The researcher investigated the decision problem of trading bitcoin and gold, examining the strategy's sensitivity to transaction costs and their effect on the outcome Ziqi.,(2022).

The researchers have talked about the value of stock price prediction in the stock market industry as well as the shortcomings of conventional approaches like technical and fundamental analysis in terms of guaranteeing consistency and accuracy in forecasts. They have concentrated on using machine learning technology, particularly LSTM and RNN, to forecast the stock market's future course J. Anusha.,(2020).

The significance of demand forecasting in the electricity industry, especially in smart grids made possible by smart metering technologies, has been covered. For power demand forecasting, they have developed an LSTM-based neural network model that outperforms conventional statistical time series analysis algorithms in terms of accuracy Koushik.,(2021).

Given the difficulties China faces in its commerce with the rest of the world development of high-precision forecasting for international trade is advantageous to China's government, guild, and export and import businesses Ye Tong.,(2023).

For regional temperature forecasting, the researchers have used deep learning techniques to surpass conventional artificial neural network architectures in terms of fewer mistakes Jenny.,(2020).

A thorough analysis of deep learning models, such as CNNs, for time series forecasting has been given by Pedro.,(2021).

After evaluating the effectiveness of several designs, they concluded that LSTM and CNN are the best options, with LSTM producing the most accurate forecasts.

An overview of previous research on COVID-19 forecasting models employing CNN-based models, including CNN, S-CNN, MA-CNN, SMA-CNN, and MAS-CNN, is presented in the work by Haviluddin.,(2023).

Convolutional layers enhance performance whereas pooling layers have the opposite impact investigation on the application of convolutional neural networks (CNNs) for the analysis of seasonal time series data with trends. It suggests utilizing a linear activation function or rectified linear unit (ReLU) in conjunction with the Adam optimizer and a non-pooling CNN (NPCNN) Shuai.,(2020).

In order to address the challenges associated with stock price prediction, including poor speed convergence, high complexity, and accurate prediction. The authors have developed a hybrid model that incorporates the Prophet and LSTM models. The back propagation neural network (BPNN) is used to further tune the model in order to enhance predicting performance Cheng.,(2022).

In order to address the challenges associated with stock price prediction, including poor speed convergence, high complexity, and accurate prediction. The researchers have developed a hybrid model that incorporates the Prophet and

LSTM models. A backpropagation neural network (BPNN) was used to further tune the model in order to enhance predicting performance Wilson.,(2023).

An extensive assessment of the literature on methods for projecting power consumption in low- and middle-income countries has been presented by Aneeque.,(2020).

An overview of machine learning models and their use in the forex market, particularly in predicting currency movement, has been given by Michael.,(2023).

The markets and stock indexes covered, as well as the kinds of variables utilized as input for forecasts, have all been the subject of research on machine learning algorithms for stock market prediction.,(2022).

Preliminaries

LSTM

LSTM networks are highly effective for time series forecasting due to their ability to capture and learn patterns in sequential data, especially when dealing with long-term dependencies. The LSTM network consists of various gates and operations to process sequential data and capture long-term dependencies.

Here's a simplified overview of the key components and equations involved in an LSTM cell:

Notation:

ht: Hidden state at time t.

ct: Cell state at time t.

xt: Input at time t.

ft: Forget gate output at time t.

it: Input gate output at time t.

c~t: Candidate cell state at time t.

ot: Output gate output at time t.

Equations:

Forget Gate: $ft=\sigma(Wf\cdot[ht-1,xt]+bf)$ (1)

Input Gate: $it=\sigma(Wi\cdot[ht-1,xt]+bi)$ (2)

Candidate Cell State: $c\sim t=\tanh(Wc\cdot[ht-1,xt]+bc)$ (3)

Update Cell State: $ct=ft\cdot ct-1+it\cdot c\sim t$ (4)

Output Gate: $ot=\sigma(Wo\cdot[ht-1,xt]+bo)$ (5)

Hidden State: $ht=ot\cdot\tanh(ct)$ (6)

Where:

Wf,Wi,Wc,Wo

These are the weight matrices for the gates of forget and input, candidate cell state, and output gate, respectively.

bf,bi,bc,bo

These are the Bias vectors for the gates forget and input, candidate cell state, and output gate, respectively.

σ: Sigmoid activation function.

tanh: Hyperbolic tangent activation function.

These Eq. 1-6 represent the flow of information through an LSTM cell. The input gate selects what fresh data to store, the forget gate selects what information from the prior cell state to delete, and the output gate determines what information to output. The candidate cell state represents the new information to be added to the cell state. The final hidden state ht carries the information to the next time step. The parameters (weights and biases) are learned during the training process.

RNN

A class of artificial neural networks called RNNs is made to process data sequentially. They can be used for time series forecasting due to their ability to capture temporal dependencies and patterns in sequential data. The calculation in a standard RNN involves a series of operations at each time step, updating the hidden state based on the input and the previous hidden state. Eq. 7 shows the formula to calculate CNN. Here's a simplified representation of the calculations in a basic RNN at a given time step t:

Notation:

ht: Hidden state at time t.

xt: Input at time t.

Whx: Weight matrix for the input-to-hidden connections.

Whh: Weight matrix for the hidden-to-hidden connections.

bh: Bias vector for the hidden layer.

tanh: Hyperbolic tangent activation function.

Equations:

Hidden State Update: $ht=\tanh(Whx\cdot xt+Whh\cdot ht-1+bh)$ (7)

Prophet

Prophet is an open-source forecasting tool developed by Facebook for time series forecasting. It is designed to handle daily observations that display patterns on different time scales, such as yearly, weekly, and daily patterns, as well as holidays and special events. Prophet is particularly well-suited for business time series data, such as sales, demand, and financial metrics. The equation for the overall model in given in Eq. 8.

t: Time index.

y(t): Observation at time t.

g(t): Trend component.

s(t): Seasonal component.

h(t): Holiday component.

ϵ(t): Error term.

Equations:

Overall Model : $y(t)=g(t)+s(t)+h(t)+\epsilon(t)$ (8)

Trend Component

Prophet uses a piecewise linear or logistic growth model for the trend. Eq. 9 is used to define the trend component:

$$g(t)=\sum_{i=1}^{N}(ki\cdot \max(0,t-ti)) \quad (9)$$

Here, N is the number of change points, ki is the rate of growth before change-point i, ti is the time of change-point i.

Seasonal component

The seasonal component accounts for weekly and yearly seasonality. It is modeled using Fourier series expansion:

s(t)=∑n=1N(an·sin(P2πnt)+bn·cos(P2πnt)) (10)

Here, N is the number of Fourier terms, P is the period (e.g., 7 for weekly, 365.25 for yearly), and an and bn are the Fourier coefficients.

Holiday component

The holiday component allows the model to account for the impact of holidays. If t corresponds to a holiday, h(t) takes a value of 1; otherwise, it is 0.

Error Term

The error term represents the difference between the observed value and the sum of the trend, seasonality, and holiday components:

$$\epsilon(t)=y(t)-(g(t)+s(t)+h(t)) \quad (11)$$

The Prophet model then uses a Bayesian approach to estimate the parameters of the model, considering the uncertainties in the observed data.

CNN

Convolutional neural networks (CNNs) are primarily known for their success in computer vision tasks, but they can also be adapted for time series forecasting. When applying CNNs to time series data, the 1D convolutional layers are particularly useful for capturing local patterns and extracting features from sequential input. The formulation of a CNN involves several components, including convolutional layers, activation functions, pooling layers, fully connected layers, and the output layer. Here's a simplified representation of the key equations for a basic one-dimensional CNN:

Notation

X: Input sequence.
Fi: Filter weights for the i-th filter.
bi: Bias term for the i-th filter.
σ(·): Activation function (commonly ReLU in CNNs).
P(·): Pooling operation (commonly max pooling in CNNs).
Wfc: Weights for fully connected layer.
bfc: Bias term for fully connected layer.
Y: Output of the network.

Convolutional Layer

Convolution Operation: Zi=σ (X∗Fi+bi) where ∗ denotes the convolution operation.

Pooling Layer

Pooling Operation: Pi=P (Zi) where P (·) is a pooling operation.

Fully Connected Layer

Flatten Operation: Flatten (Pi)=Flatten(Pi)

Fully Connected Operation

Y=σ (Flatten (Pi)·Wfc+bfc)

Output Layer (for Regression)

Output for Regression: Y^=Y (Linear activation is often used for regression tasks.)

Output Layer (for Classification)

Output for Classification: Y^=Softmax (Y) (Softmax activation is commonly used for multi-class classification tasks.)

Back-propagation

The gradients of the loss with respect to the model parameters are computed using back-propagation.

Parameter Update

The model parameters (filter weights, biases, fully connected layer weights, and biases) are updated using an optimization algorithm (e.g., stochastic gradient descent). These equations provide a high-level overview of the operations performed in a basic CNN. The actual implementation may involve additional details, such as regularization techniques, batch normalization, and specific configurations based on the architecture and task requirements.

TFT

Temporal fusion transformer (TFT) is an algorithm designed for time series forecasting. TFT leverages a transformer architecture to model complex temporal relationships in time series data and is designed to handle multi-horizon forecasting, meaning it can simultaneously predict multiple future time points.

Methodology

The methodology for carrying out this work is discussed in detail in this section.

Data Collection

The dataset utilized in this work originates from Pecan Street, Texas, USA, focusing on the average energy consumption in a smart home derived from the smart grid. The energy consumption data of various smart appliances within a smart home is recorded every 15 minutes. Subsequently, the monthly average energy consumption in kilowatts (kw) is calculated. Texas experiences four distinct seasons: summer, winter, autumn, and spring. The decomposition of the data is shown in Figure 1, which indicates that the data taken for this work is time series data.

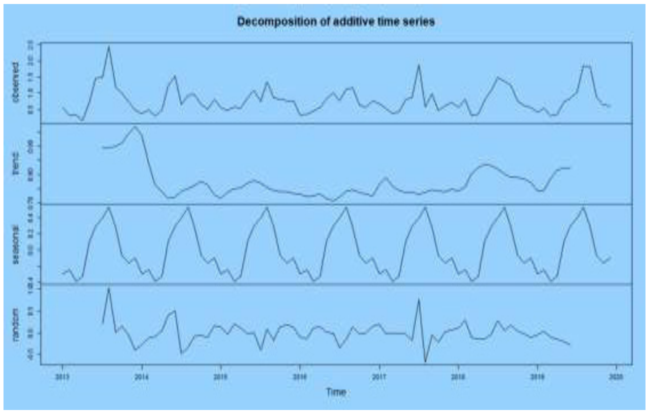


Fig. 1: Decomposition of the dataset

Data Preprocessing

Data cleaning and normalization were performed to handle missing values and ensure consistency across the dataset. Temporal features, such as time and date, were extracted to facilitate time series analysis. Spatial features were identified and extracted for use in the CNN model.

Model selection

CNN is utilized for extracting spatial features from the dataset. RNN and LSTM were employed to capture temporal dependencies inherent in time series data. A temporal fusion transformer (TFT) is applied for its capabilities in handling complex temporal patterns. Prophet is included for its proficiency in managing seasonality and holiday effects.

Model Training

The dataset was split into training and testing sets to evaluate the models' performance effectively. Each machine learning algorithm was trained on the training set, optimizing for accurate energy consumption forecasting.

Performance Metrics

Mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE) were selected as performance metrics. These metrics were employed to assess the accuracy of each forecasting model against the ground truth data in the testing set.

Comparative Analysis

Experimental results were analyzed to understand the strengths and shortcomings of each algorithm in predicting energy consumption patterns. Insights gained from the comparative analysis were used to evaluate the suitability of machine learning algorithms for forecasting energy consumption in smart homes.

Validation

The findings were validated through statistical significance testing to ensure the robustness of the results.

Implications

The results of this research work were utilized to provide insights into selecting appropriate machine learning algorithms for energy consumption forecasting in smart homes. Recommendations for enhancing energy management systems in smart homes were derived from the findings, contributing to sustainable and efficient energy usage in residential environments. The following is the pseudo-code for the above methodology:

Step 1: Bring in the needed libraries
Step 2: Load smart grid dataset from Pecan Street, Texas
Step 3: Split dataset into training and testing sets
Step 4: Define machine learning models
Step 5: Train machine learning models on the training set
Step 6: Make forecasting on the test set
Step 7: Evaluate the performance of each model using metrics
Step 8: Display performance metrics for each model
Step 9: Analyze and discuss the experimental results
Step 10: Provide conclusions and recommendations based on the findings.

RESULTS

This work uses various machine learning algorithms to forecast the average energy consumption of a smart home from the smart grid within specific timeframes. Utilizing data from the Pecan Street dataset, the collected information is partitioned into training and test sets. The outlined methodology involves meticulous data division and application of the machine learning models CNN, RNN, LSTM, Prophet, and TFT. The accuracy of each model is evaluated through comprehensive metrics, including RMSE, MAE, and MAPE.

The graphical representation of the accuracy of CNN is given in Figure 2. A validation curve typically illustrates how a model performs on a validation set. It also helps to identify the hyperparameter value that optimizes the model's performance without overfitting or underfitting. A testing curve shows how the model performs on a separate test set over different epochs or iterations during training. It also helps to evaluate how well the model generalizes to new, unseen data as training progresses. The results obtained by using various machine learning models are represented graphically. The x-axis represents different values of a hyperparameter, such as learning rates, regularization strengths, or model complexities. The x-axis represents the number of training epochs or iterations or the year based on the machine learning model that is taken for this work. The y-axis represents the performance metric, such as accuracy or error or the amount of energy consumption in kw. Figure 2 represents the graphical representation of the result obtained while using CNN. The number of epochs constitute the x-axis and the accuracy is represented in the

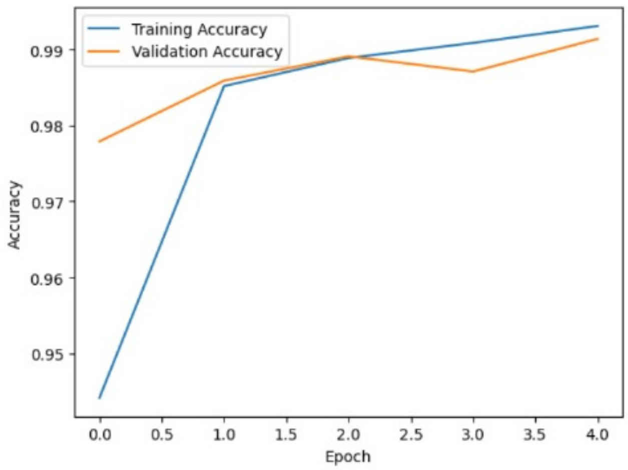
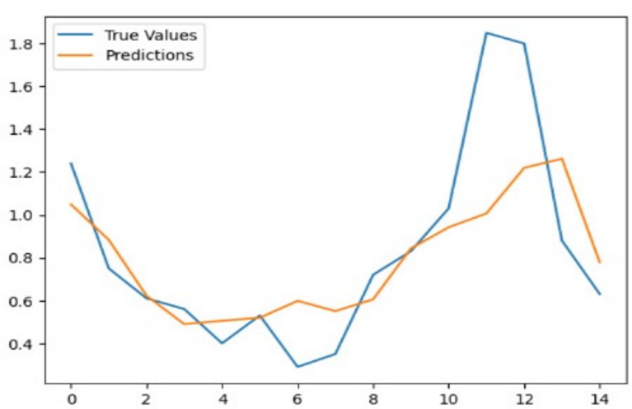


Fig. 2: Graphical representation of the output of CNN

y-axis. The graph shows that the accuracy of CNN is .99 which is a good result.

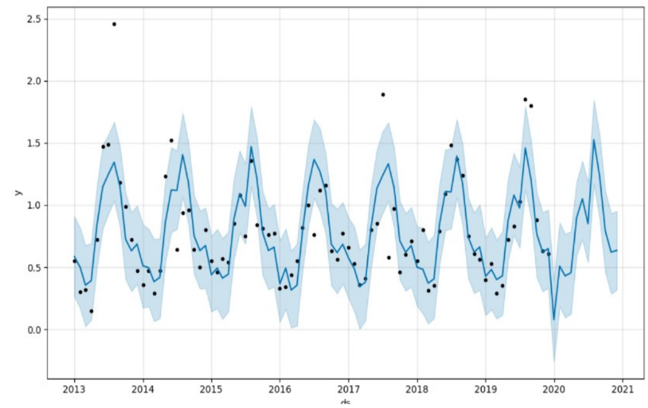
The graph in Figure 3 illustrates the validation and testing curves resulting from the utilization of RNN on the provided dataset. Notably, the true value curve experiences an increase before the predictions curve does. At specific epochs, there is a close alignment between the true value curve and the predictions curve. The y-axis denotes the average energy consumption in kilowatts, while the x-axis signifies the number of epochs, showcasing the evolution of the model's performance over the training iterations.

Figure 4 illustrates the outcomes derived from the application of the Prophet algorithm. The horizontal axis corresponds to the year, while the vertical axis depicts the average energy consumption in a smart home sourced from the smart grid. A discernible pattern emerges from the graph, as the majority of data points align along a linear trajectory, with only a small number of exceptions. This observation strongly indicates that the Prophet model is well-suited to the dataset employed in this research. The overall alignment of data points along the line underscores



x-axis: No.of Epochs y-axis: energy consumption

Fig. 3: Graphical representation of the output of RNN



x-axis : year y-axis: energy consumption

Fig. 4: Graphical representation of the output of Prophet

the model's effectiveness in accurately capturing and predicting energy consumption patterns within the context of the study.

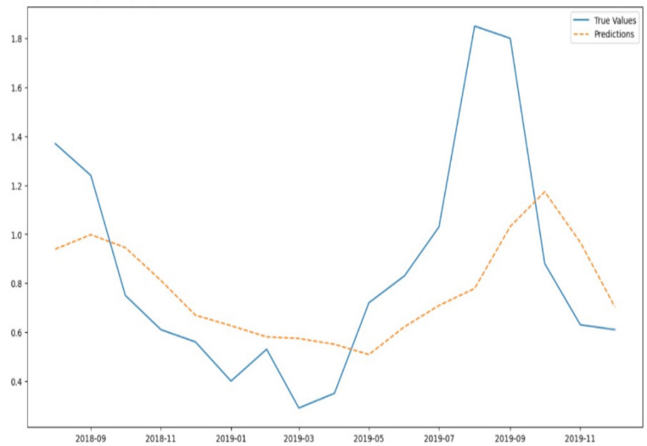
The performance graph of the NBeats model is displayed in Figure 5. Upon examination of the test values and prediction curve, it becomes evident that the model is exhibiting over-fitting. An increase in the test values indicates this, while the validation curve does not show a substantial rise. The discrepancy between the rising test values and the relatively stable validation curve suggests that the model may not be adequately capturing the complexities present in the data, leading to suboptimal generalization.

Figure 6 depicts the visual representation of outcomes generated through the application of the TFT algorithm. The horizontal axis corresponds to the year, while the vertical axis represents the average energy consumption in a smart home derived from the smart grid. A notable observation from the graph is the divergence between the true values curve and the predictions curve; they do not align along the same trajectory. This discrepancy strongly suggests that the model is experiencing overfitting, potentially due to insufficient data or the complexity of the model surpassing the requirements for forecasting this dataset.

The performance metrics like RMSE, MAE and MAPE measure the accuracy of the various machine learning models taken in this work. The values of all these metrics are given in Table 1. The comparative graph of the metrics RMSE, MAE and MAPE are given in Figures 7 and 8, respectively.

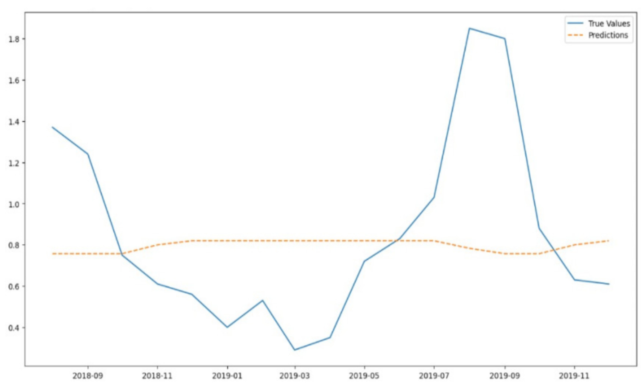
Discussions

The presented results provide a comprehensive evaluation of the forecasting performance across various models, namely RNN, Prophet, NBEATS, TFT, and LSTM. Each model's efficacy is gauged through key performance metrics: RMSE, MAE, and MAPE. Notably, RNN and Prophet emerge as strong contenders, exhibiting superior performance with an RMSE of 0.31 and 0.25, MAE of 0.21 and 0.17, and MAPE



x-axis: yeary-axis:energy consumption

Fig. 5: Graphical representation of the output of NBEATS



x-axis: year y-axis: energy consumption

Fig. 6: Graphical representation of the output of LSTM

Table 1: Metrics comparison of the forecasting models

Metrices	*RNN*	*Prophet*	*Nbeats*	*TFT*	*LSTM*
RMSE	0.31	0.25	0.39	0.48	0.44
MAE	0.21	0.17	0.31	0.36	0.32
MAPE	27.36	22.87	37.59	50.8	44.05

of 27.36% and 22.87%, respectively. These metrics signify the models' adeptness in minimizing prediction errors and accurately forecasting energy consumption patterns. In contrast, NBEATS, TFT, and LSTM demonstrate slightly higher errors, with NBEATS displaying an RMSE of 0.39, TFT at 0.48, and LSTM at 0.44. The corresponding MAE values for NBEATS, TFT, and LSTM are 0.31, 0.36, and 0.32, while their MAPE values stand at 37.59, 50.88, and 44.05%, respectively. These results illuminate the nuanced trade-offs between models, providing valuable insights for selecting an optimal approach based on specific forecasting requirements and dataset characteristics. While Prophet and RNN exhibit favorable outcomes in this evaluation, the most suitable

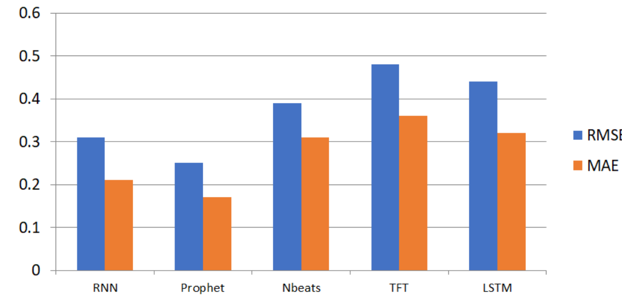


Fig. 7: Comparison of the various machine learning models

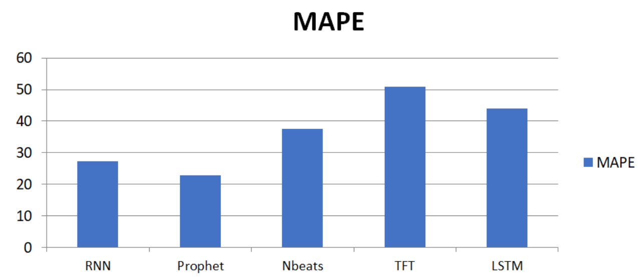


Fig. 8: MAPE of the different machine learning forecasting models

model choice ultimately hinges on the intricacies of the forecasting task and the prioritization of minimizing errors and ensuring robust predictive capabilities.

Conclusion

In conclusion, this work delves into the crucial task of forecasting energy consumption within the dynamic realm of smart homes, leveraging machine learning algorithms, including CNN, RNN, LSTM, TFT, and Prophet. The investigation, based on high-frequency energy consumption data from Pecan Street, Texas, underscores the significance of accurate predictions for optimizing resource utilization and promoting energy efficiency. Through rigorous comparative analysis using performance metrics such as MAE, RMSE, and MAPE, the strengths and limitations of each algorithm are meticulously unveiled. The experimental results showcase the commendable accuracy of CNN and Prophet models, with lower RMSE, MAE, and MAPE values compared to other models. These insights provide valuable guidance not only in selecting the most suitable algorithm for energy consumption forecasting but also in refining energy management systems within smart homes. By contributing to the advancement of sustainable and efficient energy usage in residential environments, this research facilitates the seamless integration of smart home technologies into the broader landscape of energy conservation and resource optimization.

Acknowledgments

I thank Pecan Street, Texas, USA, for providing me with the dataset to carry out my research.

References

A. Aneeque, Mir, Mohammed, Alghassab., Kafait, Ullah, A. Zafar, Khan, Yuehong, Lu, Muhammad and Imran, "A Review of Electricity Demand Forecasting in Low and Middle Income Countries: The Demand Determinants and Horizons", Sustainability, 12(15), 2020, doi: 10.3390/SU12155931.

Aditi Singh and LavnikaMarkande, "Stock Market Forecasting Using LSTM Neural Network", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2023, doi: 10.32628/cseit23903138.

Cheng, Long., Chengbo, Yu, Tao and Li, "Prophet-Based Medium and Long-Term Electricity Load Forecasting Research", Journal of Physics: Conference Series, Vol. 2356, 2022, doi: 10.1088/1742-6596/2356/1/012002.

Haviluddin, Haviluddin, Rayner and Alfred, "Multi-Step CNN Forecasting for COVID-19 Multivariate Time-Series", International Journal of Advances in Intelligent Informatics, Vol. 9, No. 02, pp. 176-186, ISSN 2548-3161, 2023, doi: 10.26555/ijain.v9i2.1080.

J. Anusha, Adhikar, K. Apeksha, Jadhav, G. Charitha, Kh. Karishma, HS. Supriya, "Literature Survey On Stock Price Prediction Using Machine Learning", International Journal of Engineering Applied Sciences and Technology, Vol. 5, Issue 8, pp. 258-262, 2020, doi:10.33564/IJEAST.2020.V05I08.040.

Jenny, Alexandra, Cifuentes, Quintero., Geovanny, Alberto, Marulanda, García, Antonio, Bello, Morales, Javier, Reneses and Guillen, "Air Temperature Forecasting Using Machine Learning Techniques: A Review", Energies, 2020, doi: 10.3390/EN13164215.

Koushik, Roy, Abtahi, Ishmam, Kazi, Abu and Taher, "Demand Forecasting in Smart Grid Using Long Short-Term Memory", In: Proc. of International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI), Rajshahi, Bangladesh, 2021, pp. 1-5, doi: 10.1109/ACMI53878.2021.9528277.

Mahinda, Mailagaha, Kumbure, Christoph, Lohrmann, Pasi, Luukka, Jari and Porras, "Machine Learning Techniques and Data For Stock Market Forecasting: A Literature Review", Expert Systems With Applications, Vol. 197, July 2022, doi: 10.1016/j.eswa.2022.116659.

Michael, Ayitey, P. Junior, Appiahene, Obed, Appiah., Christopher, Ninfaakang and Bombie," Forex Market Forecasting Using Machine Learning: Systematic Literature Review And Meta-Analysis", Journal of Big Data, Vol.10, 2023, doi: 10.1186/s40537-022-00676-2.

Pedro, Lara-Benítez, Manuel, Carranza-García., C. José, and Riquelme, "An Experimental Review on Deep Learning Architectures for Time Series Forecasting", International Journal of Neural Systems, Vol. 31, No. 03, 2021, doi: 10.1142/S0129065721300011.

Shuai, Liu, Hong, Ji, C. Morgan and Wang," Nonpooling Convolutional Neural Network Forecasting for Seasonal Time Series With Trends", in IEEE Transactions on Neural Networks and Learning Systems, Vol. 31, No. 8, pp. 2879-2888, Aug. 2020, doi: 10.1109/TNNLS.2019.2934110.

Sumriti, Ranjan, Patra., Hone, Jay, Chu and Tatas, "Regional Groundwater Sequential Forecasting Using Global And Local LSTM Models", Journal of Hydrology: Regional Studies, 2023, doi: 10.1016/j.ejrh.2023.101442.

Wilson and Paroschi, "Stock Price Forecasting Using Hybrid Prophet—LSTM Model Optimized by BPNN". Algorithms for Intelligent Systems, 2023, doi: 10.1007/978-981-19-8742-7_22.

Ye Tong, "Literature Review in International Trade Forecasting Based In Machine Learning Method", SHS web of conferences, pp. 1-4, 2023, doi: 10.1051/shsconf/202315403025.

Ziqi, Dai, Tianle, Shen, Yu, Hou, Xinxing, Luo, Suping and Zhang, "Market Trading: LSTM-Based Forecasting And Decision Making", Highlights in Business, Economics and Management, 2022, doi: 10.54097/hbem.v4i.3538.