-

Kwame Nkrumah University Of Science And Technology

College of Engineering

Short-Term Load Forecasting

Supervisor :

Contents

List of Figures

1 Introduction

In our modern society, electrical energy has become one of the main resources driving the economic and social development of any nation. It is widely used in various fields, such as production and manufacturing, mining and construction, agriculture, textiles, and communication, that the inability to provide quality and continuous power supply would lead to several issues, the least of which is a lack of faith in the Ghanaian market as a good investment ground [1]. The demand for electricity has increased with new developments in the world. Consequently, better quality and more affordable power supply is needed to sustain the current level of development [2]. This means that consumers need more economical and increasingly reliable electrical energy. At present, there is no substantial energy storage in the generation, transmission and distribution system [3], it is expected that the generation and demand of electrical energy be balanced. The generation of electricity needs to change to always meet the changing nature of demand; otherwise, the stability of the power system could be endangered [4]. In order to keep the balance of the electrical power network, a precise load prediction scheme is essential and its importance cannot be understated.

Electric load forecasting is the process of predicting the demand for electricity in a given area or region. Load prediction or forecasting can be categorized into three parts: short-term forecasting, medium-term forecasting, and long-term forecasting, based on the forecasting duration. Short-term load forecasting is the basis for power system operation and analysis, referring to the power load prediction for the next few hours, a day, or several days. This type of load forecasting is beneficial for optimizing the operation time of generating units, i.e., the starting and stopping time, and their total expected output. A precise load forecast is helpful for maximising the total consumption of the generating units [5]. Therefore, improving the accuracy of short-term load forecasting is crucial in the operation and management of the modern power system. Without it, problems like Load shedding, partial or complete shutdowns, under-voltage, etc. which causes considerable damage to equipment and human life, would become commonplace.

In the past couple of decades, the world has slowly realized the dangerous dependence it has on fossil fuels. But even with a looming future increase in energy demand, she has chosen to pivot more and more to renewable energy (RE) [6]. With the large-scale continuous integration of new energy sources into the energy mix of various countries and regions, significant changes have taken place in the power grid pattern and power supply structure, and the performance of grid has also undergone profound changes. It has brought new challenges to the safe and stable operation of the power system. For the power system, reasonable power planning is a necessary condition to ensure the stable operation of the society. Considering the intermittent nature of power supply, this becomes even more necessary. Accurate load forecasting is the basis for realizing the continually safe operation of the generating units at the start of the generation cycle. Also, a problematic characteristic of renewable energy is their stochasticity [7]. The higher the penetration of renewable energy, such as wind and solar energy in our energy mix, the larger the stochasticity of our energy supply due to the stochastic nature of REs. Therefore there is a significant requirement for accurate decision making to stabilize the demand and supply of electricity.

Load forecasting also places a financial yoke on the entire supply chain if not handled properly. Proper Economic operation and management of the power system requires an innate knowledge of the current demand so that it can be matched effectively. It is of great significance to optimize the combination of units online, economic dispatching, optimal power flow and power market transactions which can be achieved with accurate load forecasting schemes.

1.1 Aims

To perform short-term load forecasting using an optimized neural network architecture. The provision of a detailed comparison between various neural network architectures would enable one to make an informed decision regarding what topology to select for a given application. Mean average error, mean square error, and mean absolute percentage error are the metrics that would be compared.

1.2 Objectives

1. Review similar scholarly papers.

2. Collect hourly load data for four consecutive years from any relevant source.

3. Use the data to develop a short-term load forecasting model based on neural network architecture optimised by meta heuristic optimization approaches.

4. Compare its performance with other neural network architectures trained or optimized with other meta heuristic algorithms.

2 Literature Review

This section involves the critical evaluation and analysis of existing research and scholarly literature on various short-term load forecasting techniques. The reviews are intended to helps us identify gaps in our existing knowledge base, synthesize and integrate findings from multiple sources, and provide us with insights into the methods and theories that have been used in previous research, which can inform the design and implementation of our project.

2.1 A combined deep learning application for short term load forecasting [8] in 2021

In this paper, Ozer et al employs transfer learning to perform short-term load prediction to deal with the problem arising due to a limited amount of data available. Load forecasting using a machine learning-fueled approach requires a large amount of data over a wide period of time because machine learning algorithms give more accurate results with more data generally [9], [10]. However, it is not always possible to obtain historical energy consumption data. A novel approach which involves using XCORR (cross-correlation of two discrete-time sequences) to find the similarity between the data to be used for transfer learning and the original data, using the most similar data to train an Adam-optimized LSTM Network and finally, a one-month energy consumption data from a different location in the US was used to test the learned model was proposed.

In tests performed without transfer learning with original data, in the worst case 1021.671, 711.703 and 22.916 values were obtained, where in the best case, 852.912, 494.496 and 13.913 values were obtained for RMSE, MAE and MAPE, respectively.

As a result of the transfer learning process with the most suitable building selected using XCORR, 736.706, 352.176 and 8.145 values were obtained for RMSE, MAE and MAPE, respectively. Therefore, according to the absence of transfer learning, 5.768% improvement was achieved in MAPE values corresponding to the relative error reduction rate of 41.45%.

2.1.1 Advantages

1. Transfer learning allows for the use of pre-trained models, which can reduce the amount of time and computational resources needed for training. This is especially beneficial in load prediction, where large datasets and complex models are often required.
2. Transfer learning allows the model to leverage knowledge learned from a related dataset, which can lead to improved accuracy in load prediction at other times or seasons.
3. Transfer learning can help the model to generalize better to new and unseen data, which is important in load prediction as the model needs to accurately predict load demand under various conditions.

2.1.2 Disadvantages

1. The pre-trained model may be too specific to the original task, leading to overfitting on the new dataset. This may be harmful if seasonal trends and holidays are not accounted for in the original model.
2. The pre-trained model may not be applicable to the specific problem domain, leading to poor performance.

2.2 Deep learning based short term load forecasting with hybrid feature selection [11] in 2022

In this paper, a deep learning based Long Short Term Memory (LSTM) Network with hybrid feature selection namely RMR-HFS-LSTM, is proposed. The RMR-HFS is a combination of filter and wrapper feature selection introduced for identifying the optimal subset of features to reduce dimensionality by reducing the feature space. It does this by identifying the irrelevant and non-informative features by finding the relationship between each feature and target feature. Similarly, the mutual information (MI) filter feature selection calculates the dependency of each feature with the target feature using the entropy based correlation measure.

In the first phase, the hourly recorded electricity load and weather dataset is cleaned and normalized to prepare the data for further processing. In the second phase, the load samples are clustered and the dissimilar samples are eliminated. Then the instance based filter feature selection RReliefF and information theoretic based mutual information filter feature selection are applied to identify the relevant features.

Consequently, the selected features of MI and RReliefF are combined together (COMFS) to find the maximal relevant features. In the third phase, the LSTM deep prediction model is built and trained with all features and the selected features of RMR-HFS-LSTM. Then, the parameters of the constructed models are tuned by using the time series cross validation on rolling basis with validation data. Finally, the performance of the prediction model RMR-HFS-LSTM with all features and selected features are compared against MLP and RNN models in terms of the error measures such as MAPE, and RMSE.

The hourly recorded electricity load and weather data of Switzerland from the period of 1st January 2008 to 31st December 2012 is utilized for prediction. The performance of the model is compared with MLP and RNN in terms of MAPE, and RMSE. As a result the LSTM with selected features of RMR-HFS outperformed other methodologies. The deep learning based LSTM with hybrid feature selection RMR-HFS improves the accuracy of the short term load forecasting by producing least MAPE and least RMSE compared to MLP, RReliefF-MLP, MI-MLP, RFE-MLP, RNN, RReliefF-RNN, MI-RNN, RFERNN, LSTM, RReliefF-LSTM, MI-LSTM and RFE-LSTM.

2.2.1 Advantages

1. Feature selection can result in improved performance by reducing the dimensionality of the input data and eliminating irrelevant or redundant features, which can improve the accuracy and efficiency of the model.
2. Feature selection can help reduce overfitting by removing irrelevant or redundant features, which can improve the model's generalization performance.
3. Feature selection can simplify the model by reducing the number of input features, making it easier to interpret and understand.
4. By reducing the dimensionality of the input data, feature selection can reduce the training time of the model.
5. Feature selection can provide better feature ranking, which can help identify the most important features for load forecasting. This can provide insights into the load forecasting process and improve decision-making.

2.2.2 Disadvantages

1. Feature selection can result in a loss of information that may be relevant for load forecasting. This could lead to less accurate predictions.
2. LSTM is already a complex model, and adding feature selection can make it even more complex. This can result in longer training times and increased computational resources.
3. The risk of overfitting is increased when using feature selection with LSTM. This occurs when the model is too complex and fits the training data too closely, resulting in poor generalization performance on new data.
4. The feature selection process may introduce selection bias, as certain features may be favored over others based on the training data. This can lead to poor generalization performance on new data.

2.3 Machine-Learning based methods in short-term load forecasting [12] in 2021

This paper by Weilin Guo et al proposed a load forecasting method based on the fusion model of Support Vector Machine (SVM), Random Forest (RF) and Long Short-Term Memory (LSTM) frameworks, together with a data preprocessing method which invovles preprocessing by the Ensemble Empirical Mode Decomposition (EEMD) algorithm for linearizing the variation of load. At this point, the fusion method built on SVM, RF and LSTM is used to perform the load forecast. The effectiveness of the proposed forecasting approach is then verified by numeric simulations based on real load data.

In terms of the forecasting accuracy, the MAPEs of SVM, RF, LSTM and the fusion algorithm are 23.4%, 3.6%, 5.3% and 2.8%, 3.6%, 5.3% and 2.8%, respectively. It can be seen that the fusion algorithm's prediction accuracy is higher than any traditional machine learning method. It can also be seen that the simulation results verify that the proposed fusion algorithm significantly improves the prediction accuracy.

2.3.1 Advantages

1. The ability of SVM to generalize to new, unseen data alleviates the errors caused by neural network's over-fitting due to the inadequacy of input data.

2. The problem of the load involving various types of changes is hindered by the EEMD Preprocessing Technique used.

2.3.2 Disadvantages

1. Combining multiple models can increase the complexity of the model, which can result in longer training times and require more computational resources.

2. Fusion models often require additional hyperparameter tuning, which can be challenging and time-consuming.

3. The risk of overfitting is increased when using multiple models, as it may be difficult to balance the contribution of each model and prevent the model from fitting too closely to the training data.

4. The scalability of a fusion model can be limited, as it may become more challenging to integrate

additional models as the complexity of the model increases.

2.4 A fuzzy theory-based machine learning method for workdays and weekends short-term load forecasting [13] in 2021

This paper proposes a novel fuzzy theory-based machine learning method for short-term load forecasting specifically for workdays and weekends. The proposed method, named the ICE-FTS-MODA-BP model, is a hybrid approach that combines improved complete ensemble empirical mode decomposition with adaptive noise (ICEEMDAN), fuzzy time series (FTS), multi-objective dragonfly algorithm (MODA) and the back propagation (BP) neural network. To eliminate negative noise and extract inner characteristics of the raw short-term load they used ICEEMDAN. The FTS is applied for short-term forecasting with its capability of dealing with non-linear problems. It has been successfully used for predicting non-linear and dynamic datasets. The preprocessed and fuzzified data is then imported into the back propagation (BP) neural network. To achieve high forecasting stability and accuracy simultaneously, the multi-objective dragonfly algorithm (MODA) is applied to optimize the parameters of the BP. The ICE-FTS-MODA-BP method can handle non-linear relationships and uncertainties more effectively than traditional methods.

To evaluate the performance of the ICE-FTS-MODA-BP method, the author conducts experiments on two datasets, one for workdays and the other for weekends, collected from a real-world power system. The experimental results show that the proposed method outperforms the traditional methods in terms of accuracy, robustness, and efficiency. The proposed short-term load forecasting model could effectively reduce risks of power generation caused by variability. Compared to conventional methods, the ICE-FTS-MODA-BP method is better able to handle non-linear relationships and uncertainty.

2.4.1 Advantages

1. By combining multiple techniques, this approach led to more accurate load forecasting as when compared to using a single method.

2. The ICEEMDAN method can effectively separate the non-linear and non-stationary characteristics of load data, making it more robust and accurate in handling different load patterns.

3. The adaptive noise reduction feature of ICEEMDAN can help reduce noise in the data, which led to more accurate load forecasting.

4. FTS can handle uncertainty and vagueness in load data and provide more accurate forecasting in situations where the data is incomplete or uncertain.

5. MODA can help find the optimal solution by simultaneously optimizing multiple objectives, which leads to an increase in both accuracy and computational complexity.

2.4.2 Disadvantages

1. The accuracy of the hybrid approach depends on the availability of relevant data, which may not always be available or accessible.

2. Training the BP neural network can be time-consuming, especially for large datasets, which can delay the forecasting process.

3. There is a risk of overfitting when using the BP neural network, which can lead to inaccurate forecasts.

4. The hybrid approach may lack interpretability, as it involves multiple techniques that are not easily explainable, making it difficult to understand how it arrived at the forecast.

2.5 Short-term individual residential load forecasting using an enhanced machine learning-based approach based on a feature engineering framework: A comparative study with deep learning methods [14] in 2022

In this paper, Ali Forootani, Mohammad Rastegar and Ashkan Sami propose an enhanced machine learning-based approach for short-term individual residential load forecasting involving a feature engineering framework that extracts relevant features from the individual load data, such as time of day, day of week, and weather conditions by detecting outliers and selecting dominant features. The identified dominant features include temperature, dew point, and energy consumption at different time intervals. The authors then use machine learning algorithms, such as random forest and gradient boosting, to predict the load for each individual household.

To evaluate the performance of the proposed method, the authors compare it with several deep learning methods, such as long short-term memory (LSTM) and convolutional neural network (CNN). The experimental results show that the proposed method outperforms the deep learning methods in terms of accuracy and computational efficiency.

2.5.1 Advantages

1. The proposed method provides interpretable features that can help individuals and utilities understand the energy consumption patterns of each household.
2. The proposed method does not sacrifice accuracy for computational complexity because it has a relatively lower computational complexity.

2.5.2 Disadvantages

1. Feature selection can result in a loss of information that may be relevant for load forecasting. This could lead to less accurate predictions.
2. The risk of overfitting is increased when using feature selection with any machine learning algorithm and even more with Random Forest and Gradient Boosting due to they using a subset of features at each split. This occurs when the model is too complex and fits the training data too closely, resulting in poor generalization performance on new data.
3. The feature selection process may introduce selection bias, as certain features may be favored

over others based on the training data. This can lead to poor generalization performance on new data.

2.6 Electrical demand aggregation effects on the performance of deep learning-based short-term load forecasting of a residential building [15] in 2021

In this paper, Ayas Shaqour, Tetsushi Ono, Aya Hagishima, and Hooman Farzaneh proposes a method involving the aggregation of the electrical demand of multiple households to improve the accuracy of short-term load forecasting. The authors use a long short-term memory (LSTM) and gated recurrent unit (GRU), to predict the aggregated load. The experimental results show that the proposed method improves the accuracy of short-term load forecasting compared to traditional and machine learning-based methods.

Furthermore, the study investigates the effects of different aggregation levels on the performance of the proposed method. The authors compare the performance of the proposed method for various aggregation levels, such as daily, weekly, and monthly aggregation. The experimental results show that the aggregation level has a significant impact on the performance of the proposed method, and that weekly aggregation provides the best performance. The study contributes to the literature on short-term load forecasting for residential buildings by investigating the effects of electrical demand aggregation on the performance of deep learning-based methods. The proposed method can provide accurate load forecasting for multiple households and can help energy management systems optimize energy consumption and reduce costs. The study also highlights the importance of considering the aggregation level when developing load forecasting models. In conclusion, the study conducted by Ayas Shaqour, Tetsushi Ono, Aya Hagishima, and Hooman Farzaneh investigates the impact of dwelling aggregations on Short-Term Load Forecasting (STLF) using deep learning and feature selection methodologies. Environmental data features are combined with electrical demand time-series data to extract multiple time-series features. The results show extremely low MAPE errors and suggest that errors below 10% can be sustained at an aggregation level of 30. The study also highlights the challenges of forecasting with lower levels of aggregation and provides insights into the five classes of state-of-the-art DL methodologies. The findings can help establish future standards for optimal performance in microgrids, local demand response markets, virtual power plants, and large building-level HRES systems.

2.6.1 Advantages

1. The paper's focus on load forecasting for residential buildings makes it relevant to real-world scenarios, where accurate load forecasting is essential for efficient energy management. The findings of the paper can be applied in practice to improve energy efficiency and reduce costs for residential buildings.

2.6.2 Disadvantages

1. The model can be computationally expensive and require significant computational resources, such as GPUs or cloud computing services. This can make the model less practical for real-time applications or for small-scale implementations.
2. The model can have complex architectures, which can make them difficult to interpret and understand. This can make it challenging for researchers and practitioners to identify the factors that contribute to the model's performance and to troubleshoot issues.
3. The model requires large amounts of data for training, which can be challenging to obtain for some applications. Additionally, the quality of the data can impact the accuracy of the model, which can be a disadvantage if the data is noisy or incomplete.

2.7 Aggregated short-term load forecasting for heterogeneous buildings using machine learning with peak estimation [16] in 2021

Amine Bellahsen and Hanane Dagdougui in this paper propose a machine learning-based method for aggregated short-term load forecasting that accounts for peak estimation for heterogeneous buildings.

Two different methods for predicting energy consumption are compared and a novel approach introduced; the first method (COSMOS), multiple short-term load forecasting models were combined using a stacking ensemble approach, outperforming many recent machine learning approaches. In the second method, a combination of recurrent neural networks and one-dimensional convolutional neural networks with an inception module was used to perform day-ahead load forecasting for three industrial distribution complexes in South Korea, outperforming other methods such as multi-layer perceptron, RNN, and 1-D CNN.

The proposed method involves aggregating the load of multiple buildings and using machine learning algorithms to predict the aggregate load. The authors use peak estimation to account for the variability in the energy consumption patterns of different buildings. Specifically, the peak estimation method involves using clustering algorithms to group buildings with similar energy consumption patterns and estimating the peak load for each group separately. To evaluate the performance of the proposed method, the authors conduct experiments on a dataset of 40 heterogeneous buildings. The experimental results show that the proposed method outperforms traditional methods in terms of accuracy and can account for the variability in the energy consumption patterns of different buildings. The results show that using the random forest algorithm and ACF with building-level data and then aggregating the results to predict the district load produces more accurate forecasts. The forecasts' reliability is measured in terms of the percentage of peak over a certain subscribed power.

2.7.1 Advantages

1. The proposed model is scalable, meaning it can be applied to multiple buildings with varying characteristics. The model aggregates the data from different buildings, and then uses machine learning to predict the total energy consumption for the entire group of buildings.

2. The model allows for real-time monitoring of energy consumption in buildings, which can be

used to identify potential issues and optimize energy usage. This can help reduce energy costs and improve energy efficiency.

2.7.2 Disadvantages

1. The proposed model requires a large amount of data, including historical energy consumption data for each building, weather data, and building characteristics. Gathering and maintaining this data can be time-consuming and costly.
2. The proposed model is designed to be scalable and applicable to a wide range of buildings, but it may not be fully customizable to meet the specific needs of each building.

2.8 Forecasting of electricity demand by hybrid ANN-PSO under the shadow of the Covid-19 pandemic [17] in 2021

Mohamed Rahmoune et al looked at the impact of the lockdown on Algerian power consumption during the COVID-19 pandemic; employing a hybrid ANN-PSO model. Electricity consumption data was applied as an input to the PSO-ANN model. A quarter-hour of four consecutive years of data from Socièté Nationale de l'Electricité et du Gaz (SONELGAZ), Algeria, were incorporated into the process. The results of the PSO-ANN prediction model had better accuracy than the ANN model. The ANN architecture consisted of 3 layers with 10–5–1 neuron. Here, in this paper, smart methods were not incorporated to predict the future but rather to show the impact of the pandemic. The Hybrid PSO-ANN algorithm was adopted to demonstrate the impact of COVID-19 on electricity consumption to demonstrate that two basic steps were used. The first step is to demonstrate that the hybrid method is effective for prediction. In the second step, we continued to use the same hybrid style after the COVID-19 emerged, that is, for 2020, to note the difference between the prediction and the current prediction, which stands for the impact of this epidemic in the short term. The STLF model was applied to a day in February and April 2019 and a MAPE of 2.01% was obtained. The same model applied to a day in 2020 gave a substantial error of 6.78% due to impact of covid-19 on energy consumption.

2.8.1 Advantages

1. The PSO-ANN prediction model outperformed the ANN model in terms of accuracy.
2. PSO-ANN algorithm method was easy to implement because of the quarter-hour prediction.

2.8.2 Disadvantages

1. The algorithm can fall into local optimum in high dimensional space.
2. The algorithm had a low convergence rate.
3. The paper does not discuss the implications of the findings for energy policy or provide recommendations for energy stakeholders.

2.9 Short-Term Load Forecasts Using LSTM Networks [18] in 2019

In this paper, the authors used both traditional and LSTM methods to analyze 12 months of load data along with exogenous variables (Irradiance, Temperature, Relative Humidity and Wind Speed). The data is processed and all exogenous variables placed into one dataset. For purposes of accuracy comparison, the authors used three statistical modelling techniques - Autoregressive Moving Average Model (ARMA), Seasonal Autoregressive Integrated Moving Average (SARIMA), and Autoregressive Moving Average eXogenous (ARMAX) trained on the same data.

In terms of the forecasting accuracy, the Mean Absolute Percentage Error (MAPE) of ARMA, SARIMA, ARMAX and LSTM models over a Forecast Horizon of 24 hours are 5.42%, 6.53%, 7.51% and 1.522%, respectively. It can be seen that the LSTM's prediction accuracy is higher than any traditional statistical learning method. It was also seen from the simulation's results that the proposed LSTM model significantly improves the prediction accuracy.

2.9.1 Advantages

1. The proposed approach can capture complex temporal patterns in the load data and handle long-term dependencies.

2. The proposed approach can handle missing or noisy data, which is common in load forecasting.

2.9.2 Disadvantages

1. LSTM networks have a high computational complexity and high training time, which can be a limitation for real-time applications.

2.10 Using deep learning for short-term load forecasting [19] in 2020

This paper introduced a novel CNN for short-term load forecasting. Chronological context (weekends and special days) and exogenous variables (daily temperature) were used as inputs for artificial intelligence models. 4 years (2013–2016) of hourly load in Algeria was provided by the Algerian National Company of Electricity and Gas (Sonelgaz) and this data was used after normalization. The data was normalized to keep it relevant as actual load values increase as the population [20] greenhouse gas emissions, gross domestic product and labour growth [21] increases. The proposed CNN used a two-dimensional input, built using historical load data with exogenous variables. The results of the 2d-CNN were compared to the 1d-CNN as well as other machine learning models (ensemble methods, SVR and ANN).

The one-step-ahead (15 min) and 1-day-ahead (24 h) forecasting accuracies were evaluated using MAPE and RMSE. For each type of architecture used, Farah et al. tweaked the internal structure of the model (i.e. number of trees for Random Forest, the values for C and v for SVM, the number of estimators for GBRT (Gradient Boosting Regression Trees), the number of hidden layers for the Artificial Neural Network and Convolution Layers, Fully Connected Layers and Pooling Layers for the Convolutional Neural Network (CNN)). The best results of the SVR were obtained by the SVM architecture which used m = 0.3, and C = 5 to obtain a MAPE of 0.911% and an Root-Mean-Square Deviation (RMSE) of 85.30MW. The RF with 100 trees gave the best results with a MAPE of 0.902% and a Root-Mean-Square Deviation (RMSE) of 85.00MW, while GBRT obtained better results of a MAPE of 0.881% and an RMSE of 83.04MW with 600 estimators. By using Artificial Neural Networks, the results were generally degraded and the best performing ANN with 25 nodes in the first hidden layer and 5 nodes in the second produced a MAPE of 1.008% and an Root-Mean-Square Deviation (RMSE) of 93.82MW. The ANN was implemented using a sigmoid activation function alongside MSE loss function, while ADAM was used for gradient optimization. The Type 2 (Has ReLu activated Convolution layers (Conv) and Sigmoid activated Fully connected layers (FC) but no Pooling layers) 2d-CNN have the best overall MAPE and RMSE of 0.808% and 75.57MW.

The above results shows that both 1d-CNN and 2d-CNN perform exceptionally well compared to other machine learning techniques.

2.10.1 Advantages

1. The paper showed that 2D CNNs are very effective at extracting relevant features from complex input data, such as time series data. By applying multiple convolutional layers to the input data, the authors realized that a 2D CNN can learn to identify important patterns and relationships in the data that may be useful for load forecasting.
2. Load forecasting can involve identifying spatial patterns across a geographical area. 2D CNNs are well-suited to identifying patterns in two-dimensional space, such as the layout of a power grid or the distribution of energy usage across a region. The feasibilty of this was proven as the RMSE was proven to be about 75MW.
3. 2D CNNs are computationally efficient and can be trained quickly on large datasets, making them well-suited to large-scale load forecasting applications.
4. Pre-trained 2D CNN models can be used as a starting point for load forecasting tasks, where they can be fine-tuned on the specific dataset at hand. This can save time and computational resources, while also improving performance.

2.10.2 Disadvantages

1. For load forecasting, temporal information is critical, and a 2D CNN may not be able to capture it as well as spatial information.
2. 2D CNNs are not be as flexible as other models in dealing with variable-length time series data resulting in poor generalization performance and inaccurate load forecasting.

2.11 Application of Particle Swarm Optimization Combined with Long and Short-term Memory Networks for Short-term Load Forecasting [22] in 2022

The learning factor, the number of iterations and the population size, as well as the velocity and position taking space are determined, i.e., the parameters are initialized to various particles to find the optimal values for the hyperparameters. Once the PSO algorithm reaches the maximum limitation on the number of iterations, the load data are processed using the LSTM model after PSO training is completed to obtain the power load forecast values.

Five evaluation metrics were used, namely Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), Mean Absolute Deviation (MAD), R-Squared coefficient of determination (R^2), and Theil IC (TIC) and the values obtained were 0.0403, 3.2549, 2.9823, 0.9386 and 0.0218.

After the comparative analysis of the prediction models, it is found that both the BP neural network prediction model and the prediction model combining BP neural network and particle swarm optimization algorithm are slightly inferior to the LSTM model corresponding to the same optimization conditions with the same input in this experimental setting. The PSO-LSTM model proposed in this paper has the best prediction effect among the models used in this experiment.

2.11.1 Advantages

1. The combination of Particle Swarm Optimization (PSO) and Long and Short-term Memory (LSTM) networks has resulted in higher accuracy in short-term load forecasting, compared to traditional methods.
2. The PSO-LSTM approach can be used to forecast load for different time horizons, from hours to days.
3. The approach can be applied to large-scale systems, making it suitable for use in real-world applications.
4. The approach is incredibly generalizable to different types of load data or different geographic locations.

2.11.2 Disadvantages

1. The PSO-LSTM approach is relatively complex, and the implementation of the algorithm may require a significant amount of computational resources.
2. The approach requires a large amount of historical load data to train the LSTM networks effectively.
3. The PSO-LSTM approach is a black-box method, making it challenging to interpret the results.

3 Theory and Design Considerations

Over the past few years, a number of conventional and AI-based techniques have been used to solve the problem of short-term load forecasting. Neural networks are powerful tools for modelling complex and nonlinear relationships between input and output variables. However, designing an optimal neural network architecture for short-term load forecasting can be a challenging task due to the large number of input variables and the complex nature of the problem. In recent years, meta-heuristic optimization techniques have been applied to effectively design neural network architectures for load forecasting. These techniques aim to find the optimal architecture that minimises the forecasting error by searching the space of possible architectures efficiently. In this section, the neural network's theory, various meta-heuristic techniques and its design are briefly described.

3.1 Neural Network Architectures

3.1.1 Artificial Neural Network

Artificial neural networks are inspired by the computational principles of the human brain and aim to replicate human activities. They consist of multiple simple processing elements that operate in parallel. The network's structure, connection strengths, and processing at nodes collectively determine its function. The neural network is composed of three layers: the input layer, hidden layer, and output layer, connected by synaptic weights.

The input layer serves as a non-processing layer, responsible for feeding data into the neural network. The hidden layer, known as the processing layer, receives the weighted summation of the input data along with the bias. It incorporates an activation function to process the information. The sigmoid function is the most commonly used activation function due to its simple derivatives and soft switching properties.

The output layer is another processing layer that employs activation functions to determine the final result of the neural network. By adjusting the synaptic weights and biases throughout the network, the neural network can learn and make predictions based on the provided data.

3.1.2 Convolutional Neural Network

Convolutional Neural Networks (CNNs) are designed based on the computational principles of the brain, aiming to understand and replicate human activities, particularly in image and pattern recognition tasks. A CNN can be described as a system comprising many simple processing elements operating in parallel, whose function is determined by the network's architecture, connection strengths, and the processing performed at computing elements or nodes.

A CNN consists of multiple layers, typically including convolutional layers, pooling layers, and fully connected layers. The input layer receives the raw image data, and the subsequent convolutional layers apply filters (kernels) to extract features from the input data. Each filter scans the input image, capturing different patterns and spatial information. The result of the convolutional operation is a set of feature maps highlighting distinctive features.

Following the convolutional layers, the pooling layers downsample the feature maps, reducing the spatial dimensions and computational complexity while retaining the most important information. Pooling is usually performed using techniques like max pooling or average pooling.

The fully connected layers are the last part of the CNN and serve as the processing layer. They take the flattened output from the previous layers and apply weights and biases to perform high-level feature combination and classification. An activation function, commonly ReLU (Rectified Linear Unit), is applied to introduce non-linearity in the model, making it capable of learning complex patterns.

The output layer, also a processing layer, consists of activation functions tailored to the specific task, such as softmax for multi-class classification or sigmoid for binary classification. The output layer's neurons produce the final predictions or classifications based on the learned representations from the previous layers.

Throughout the training process, the CNN learns the optimal values for the filter weights and biases, adjusting its internal parameters to minimize the prediction errors. This learning process enables the CNN to make accurate predictions on new, unseen data, thus replicating human-like recognition capabilities for image-related tasks.

3.1.3 Long Short Term Memory Neural Network

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) designed to capture long-range dependencies and sequential patterns in data, making them particularly effective for tasks involving time-series or sequential data. LSTMs are inspired by the computational principles of the brain and aim to understand and replicate human-like memory and learning abilities.

An LSTM can be described as a system comprised of many simple processing elements operating in parallel, where the function is determined by the network's architecture, connection strengths, and the processing performed at computing elements or nodes.

The core building block of an LSTM is the LSTM cell, which consists of multiple gates that regulate the flow of information within the cell. These gates include the input gate, forget gate, and output gate. The input gate determines how much of the new input data should be stored in the cell's memory, while the forget gate decides which information is no longer relevant and should be discarded from the memory. The output gate controls how much of the cell's memory should be used to generate the output at each time step.

LSTMs can process sequences of data over time, where each time step corresponds to a specific element in the sequence. At each time step, the LSTM cell takes an input, the current element in the sequence, and the hidden state from the previous time step. The cell processes this information through its gates, updating its internal state and producing the output for that time step.

The LSTM's ability to maintain and update its internal memory over long periods allows it to capture dependencies between distant elements in the sequence, making it well-suited for tasks involving context and temporal dynamics. This memory retention capability helps LSTMs avoid the vanishing gradient problem often encountered in traditional RNNs, which struggle to retain information over long sequences.

LSTMs are widely used in various applications, such as natural language processing, speech recognition, sentiment analysis, and time-series prediction. Through the training process, the LSTM learns the optimal parameters for its gates and connections, enabling it to effectively learn patterns and dependencies in the sequential data and make accurate predictions or classifications based on the learned information.

3.2 Proposed Neural Network Architecture

The proposed architecture, a hybrid CNN-LSTM for short-term load forecasting, consists of the following components:

1. **Input Layer**: The input layer receives the sequential data containing the historical load information. In this case, the input layer has 4 time steps, representing the load of the past four weeks, and each time step contains **9** features. These features include
   - the load of the previous week,
   - load of two weeks back,
   - load of three weeks back,
   - load of four weeks back,
   - average load, day of the week,
   - weekday or weekend indicator,
   - holiday indicator, and
   - time/hour of the day.
2. **CNN Layers**: 1-Dimensional Convolutional layers (CNN) were added to the model to extract spatial features from the input data. The CNN layers apply filters to the input sequences, capturing relevant patterns and spatial information. This allows the model to focus on local features and identify important load-related patterns in the input data.
3. **LSTM Layers**: Following the CNN layers, LSTM layers are incorporated as the core processing units in the network. The LSTM layers capture temporal dependencies and patterns in the data.
4. **Output Layer**: The output layer is a single neuron that produces the forecasted short-term load value. Since the objective is short-term load forecasting, only one neuron is sufficient to provide the predicted output.

The explanation of our hybrid **CNN-LSTM Model** is given below. The main objective was to forecast short-term load demand by leveraging the powerful capabilities of the proposed neural network architecture.

3.2.1 Data Preprocessing

We began the process by performing data preprocessing on the input data. The dataset was loaded into a Pandas DataFrame, and the features were extracted by excluding the `datetime` and `DEMAND` columns. To ensure that the features were on a similar scale, we applied Min-Max scaling using the `MinMaxScaler` from scikit-learn. The dataset was then transformed into input sequences of fixed length, where each sequence contained four time steps and nine features. The target values were extracted accordingly. The data was further split into training and test sets with an 80-20 ratio.

3.2.2 Explanation of Model Architecture - CNN-LSTM Model

The neural network architecture used for short-term load forecasting consisted of a combination of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) layers. The model's structure is as follows:

1. **Input Layer**: The input layer accepts sequences of four time steps, each containing nine features representing historical load information.

2. **Conv1D Layer**: A one-dimensional convolutional layer with 64 filters and a kernel size of 4 is applied to capture spatial patterns in the input sequences.

3. **Dropout Layer 1**: A dropout layer with a dropout rate of 0.2 is added to reduce overfitting during training.

4. **Bidirectional LSTM Layer 1**: The bidirectional LSTM layer with 128 units and a hyperbolic tangent (tanh) activation function is used to model temporal dependencies in both forward and backward directions.

5. **Dropout Layer 2**: Another dropout layer with a dropout rate of 0.2 is employed to enhance generalization.

6. **Dropout Layer 3**: A dropout layer with a dropout rate of 0.2 is added after the attention module.

7. **Layer Normalization**: To ensure stable training, layer normalization with a small epsilon value is applied.

8. **Bidirectional LSTM Layer 2**: Another bidirectional LSTM layer with 128 units is utilized to further capture complex temporal patterns.

9. **Dropout Layer 4**: A dropout layer with a dropout rate of 0.2 is employed once again to mitigate overfitting.

10. **LSTM Layer 3**: The final LSTM layer with 128 units and a hyperbolic tangent (tanh) activation function is applied to generate the model's output.

11. **Output Layer**: The output layer is a dense layer with one neuron and a linear activation function to produce the forecasted short-term load demand.

3.2.3 Model Compilation and Evaluation

The model is compiled using the Adam optimizer, and the mean absolute percentage error (MAPE) is chosen as the loss function. During training, the model learns from the training dataset to make accurate predictions for unseen test data. To assess the model's performance, we monitored the MAPE on both the training and test datasets.

The choice of architecture is motivated by the characteristics of the data and the objective of the forecasting task. By combining the CNN layer with the LSTM layers, the model effectively captures both spatial and temporal dependencies in the sequential load data. The CNN layer helps identify local patterns and spatial correlations, while the LSTM layers focus on temporal patterns and long-range dependencies, resulting in improved forecasting accuracy.
This LSTM-based architecture with an added CNN layer aims to provide accurate and efficient short-term load forecasts, making it suitable for real-world forecasting scenarios where both spatial and temporal information play crucial roles in load prediction.

4 Methodology

This chapter describes how the project will be carried out. It details a step-by-step approach to the project. The programming language used for our project was Python. Python has a user-friendly interface with a lot of in-built libraries and an active community that have written lots of numerical analysis and machine learning libraries to help to effectively carry out artificial intelligence applications. After attaining the Panama Power System dataset, the following steps were taken:

1. Acquisition of Data
2. Data Pre-processing
3. Technique for Pre-processing the data - pick one to pre-process the data. See if this affects the values got.
4. Exploratory Data Analysis
5. Correlation Analysis
6. Feature Selection
7. Normalization of fields' values
8. Hyperparameter Tuning
9. Optimization of hyperparameters by PSO
10. Optimization of weights and biases by Modified Back Propagation

4.1 Acquisition of Data

The source of the data was the Panama Energy Company. The raw dataset was obtained from Kaggle. The dataset contains hourly records from January 2015 until June 2020. The structure of the data is as below;

4.1.1 Structure of the Data

The structure of the Panama Energy Company's dataset is as follows;

Column name	Description	Unit
datetime	Date-time index corresponding to Panama time-zone UTC-05:00 (index)	
week_X-2	Load lag from second previous week before forecast	MWh
week_X-3	Load lag from third previous week before forecast	MWh
week_X-4	Load lag from fourth previous week before forecast	MWh
MA_X-4	Load lag moving average, from first till fourth previous weeks before forecast	MWh
dayOfWeek	Day of the week, starting on Saturdays	[1,7]
weekend	Weekend binary indicator	1 = weekend, 0 = weekday
holiday	Holiday binary indicator	1 = holiday, 0 = regular day
Holiday_ID	Unique identification number	integer
hourOfDay	Hour of the day	[0, 23]
T2M_toc	Temperature at 2 meters in Tocumen, Panama city	ºC
DEMAND	National electricity load (Target or Dependent variable)	MWh

Figure 1: Fields of the Panama Energy Consumption dataset

datetime	week_X-2	week_X-3	week_X-4	MA_X-4	dayOfWeek	weekend	holiday	Holiday_ID	hourOfDay	T2M_toc	DEMAND
2015-01-31 01:00:00	962.2865	906.958	970.345	938.00485	1	1	0	0	1	25.30849609	954.2018
2015-01-31 02:00:00	933.3221	863.5135	912.1755	900.284075	1	1	0	0	2	25.14144287	913.866
2015-01-31 03:00:00	903.9817	848.4447	900.2688	881.704325	1	1	0	0	3	25.00673828	903.3637
2015-01-31 04:00:00	900.9995	839.8821	889.9538	876.458825	1	1	0	0	4	24.89971313	889.0806
2015-01-31 05:00:00	904.3481	847.1073	893.6865	879.190775	1	1	0	0	5	24.82155762	910.1472

Figure 2: Structure of the Raw Data

4.2 Data Pre-processing

Data pre-processing is a critical step in preparing the historical load data for training the LSTM-based architecture for short-term load forecasting. This section describes the various steps undertaken to clean, normalize, and transform the data to make it suitable for the model.

4.2.1 Data Cleaning

The first step in data pre-processing involves checking for and handling missing or incomplete data. In our case, we conducted a thorough examination of the dataset to identify any missing values. Missing values were either imputed using appropriate techniques or the corresponding data points were removed, depending on the extent and nature of the ”missing-ness”. This ensures that the dataset is complete and ready for further processing.

4.2.2 Feature Scaling

To facilitate convergence during model training and ensure all features contribute equally to the forecasting process, we applied feature scaling. Here, we utilized the Min-Max scaling technique to normalize the numerical features in the dataset to a common range (typically [0, 1]). This normalization ensures that features with larger magnitude do not dominate the model's learning process.

4.2.3 Sequence Creation/Enabling

Since we are dealing with sequential data, we created and checked input sequences and their corresponding target values for training the LSTM-based model. For each time step, we formed sequences of four weeks of load data, which align with the architecture's input layer configuration. We then associated each sequence with its target value, representing the load for the next time step. This sequence creation allows the model to learn temporal dependencies and patterns in the data.

4.2.4 Input Reshaping

As the LSTM layers require a 3D input tensor of the form (samples, time steps, features), we reshaped the input sequences accordingly to match this format. The sequences were structured as 3D arrays, with each sample representing a time series and containing multiple time steps with their respective features.

After completing the data pre-processing steps outlined above, the dataset is ready for training.

4.3 Exploratory Data Analysis (E.D.A.)

Exploratory Data Analysis (EDA) is a crucial step in understanding the characteristics and patterns present in the historical load dataset. In this section, we present an overview of the data through visualizations and statistical summaries to gain insights into its structure and identify potential relationships between the variables.

4.3.1 Summary Statistics

We begin the analysis by providing summary statistics for the numerical features in the dataset. This includes measures such as mean, standard deviation, minimum, maximum, and quartile values. These statistics offer a quick overview of the central tendencies and dispersions of the load data, giving an initial understanding of its distribution.

4.3.2 Time Series Visualization

Since the data consists of sequential load information, we plot time series visualizations to observe any patterns or trends over time. The time series plot allows us to identify seasonal variations, weekly patterns, and any long-term trends present in the historical load data.

4.3.3 Feature Distributions

Next, we explore the distributions of individual features using histograms and density plots. This helps us understand the range and spread of each feature and identify potential outliers or skewed distributions.

4.3.4 Correlation Analysis

To uncover relationships between different features, we conduct correlation analysis. We calculate the correlation coefficients between load and other relevant features, visualizing them through heatmaps or scatter plots. Positive or negative correlations provide valuable insights into which variables might have significant impacts on load forecasting.

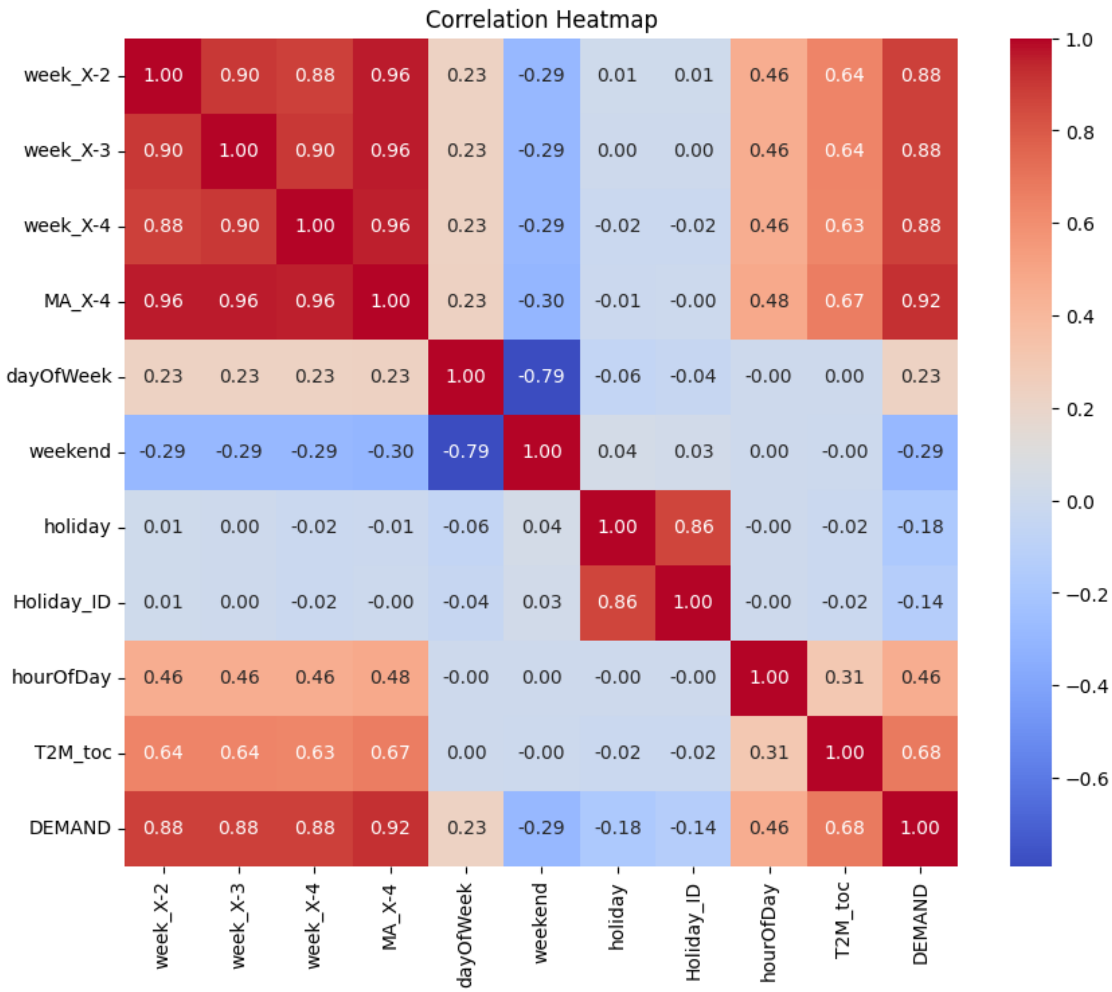

Figure 3: Correlation of every field with all other fields in the dataset

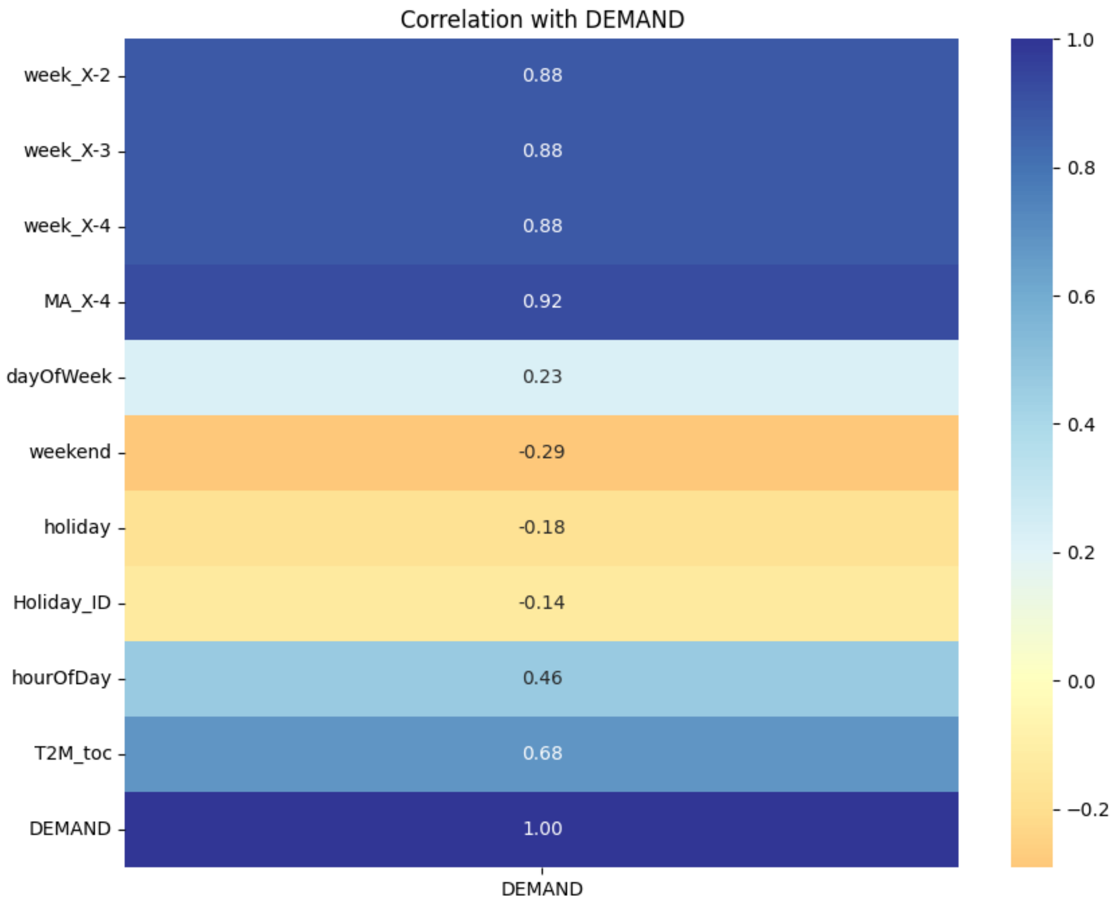

Figure 4: Correlation of DEMAND with all other fields in the dataset

The exploratory data analysis provides valuable insights into the historical load data, helping us make informed decisions about model selection and feature engineering. The findings from this section guide the subsequent steps in the short-term load forecasting process using the LSTM-based architecture with the added CNN layer.

4.4 Normalization of Fields' Values

In this section, we describe the normalization process applied to the input features before feeding them into the neural network model. Normalization is a crucial preprocessing technique used to scale the features to a common range, usually between 0 and 1 or with a mean of 0 and standard deviation of 1. This step is essential to avoid biases in the training process and to ensure that all features contribute equally to the model's predictions.

4.4.1 Min-Max Normalization

The Min-Max normalization method was employed to scale the numerical features in the dataset. For each feature, we transformed its values using the following formula:

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

where X is the original feature value, X_{min} is the minimum value of the feature in the dataset, and X_{max} is the maximum value of the feature. This formula scales all values to fall within the range [0, 1].

4.4.2 Standardization

In addition to Min-Max normalization, standardization was applied to certain features. Standardization, also known as z-score normalization, transforms the features to have a mean of 0 and a standard deviation of 1. This process is particularly useful when features have different units or magnitudes. The standardization formula for each feature is given by:

$$X_{\text{std}} = \frac{X - \mu}{\sigma}$$

where X is the original feature value, μ is the mean of the feature in the dataset, and σ is the standard deviation of the feature.

4.4.3 Categorical Feature Encoding

For categorical features, such as ”Day of the Week” and ”Weekday/Weekend,” one-hot encoding was applied to convert them into numerical representations suitable for the neural network model.

4.4.4 Reasons for Normalization

The normalization process ensures that all input features are on a consistent scale, which facilitates the training process and prevents certain features from dominating the model’s learning. Moreover, it can lead to faster convergence and better performance of the neural network, especially in scenarios where features have significantly different ranges.

By normalizing the fields' values, we ensure that the LSTM-based neural network can effectively learn from the data and make accurate short-term load forecasts.

5 Results and Analysis

In this chapter, the results of the project carried out thus far are presented as well as a detailed discussion of the results. The comparison of the various neural network architectures and the proposed architecture was based on mean absolute percentage error using the same number of training dataset and test dataset. The results covered:

- Correlation Analysis results
- Results on training dataset
- Results on test dataset
- Comparison of the proposed architecture and other architectures' mean absolute percentage error values
- Comparison of the proposed architecture and other architectures from literature's mean absolute percentage error values

5.1 Results

5.1.1 Correlation Analysis results

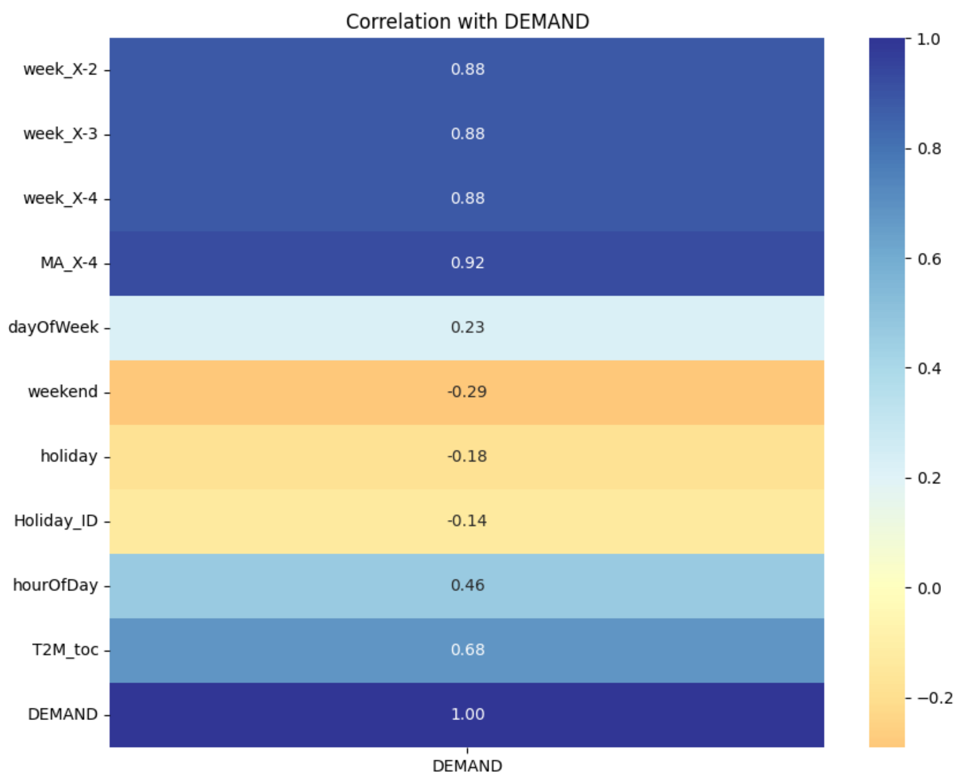

Figure 5: Correlation of DEMAND with all other fields in the dataset

The results from this analysis portrays the following deductions:

- *MA_X-4* field which represents the Load lag moving average, from first till fourth previous weeks before forecast (MWh) has the highest positive correlation of **0.92** with the output field (DEMAND).

- *week_X-2* (Load lag from second previous week before forecast (MWh)), *week_X-3* (Load lag from third previous week before forecast (MWh)), and *week_X-4* (Load lag from fourth previous week before forecast (MWh)) all have a **0.88** positive correlation with the output field (DEMAND).

- *T2M_toc* (Temperature at 2 meters in Tocumen, Panama city (ºC)) has a positive correlation of **0.68** with the output field (DEMAND).

- *hourOfDay* (Hour of the day which is an integer belonging to the closed range - [0, 23]) has a positive correlation of **0.46** with the output field (DEMAND).
- *dayOfWeek* (Day of the week, starting on Saturdays which is an integer belonging to the closed range - [1,7]) has a weak positive correlation of **0.23** with the output field (DEMAND).
- *weekend* represents a Weekend binary indicator where 1 = weekend, 0 = weekday. It has a negative correlation of **0.29** with the output field (DEMAND).
- *holiday* represents a Holiday binary indicator where 1 = holiday, 0 = regular day. It has a negative correlation of **0.18** with the output field (DEMAND).
- *Holiday_ID* represents a Unique identification number integer. It has the lowest negative correlation of **0.14** with the output field (DEMAND).

5.1.2 Results of the Hybrid Model on the training dataset

These are the results of the Model; the CNN-LSTM model consists of a total of 490,177 parameters, all of which are trainable.

Parameters	**CNN-LSTM Model**
Number of training samples	36720
Activation functions - Convolutional Layer	ReLU
Activation functions - Bidirectional LSTM Layer	Tanh
Activation functions - Output Layer	Linear
Epochs	350
Mean Absolute Percentage Error	2.7249

5.1.3 Results of the CNN-LSTM Model on testing dataset

These are the results of the hybrid model on the testing dataset.

Parameters	**CNN-LSTM Model**
Number of training samples	168
Activation functions - Convolutional Layer	ReLU
Activation functions - Bidirectional LSTM Layer	Tanh
Activation functions - Output Layer	Linear
Mean Absolute Percentage Error	2.7378

5.1.4 Results of Hybrid CNN-LSTM on testing dataset with other Models

These are the results of the hybrid CNN-LSTM model on the testing dataset.

Model	**MAPE Results (%)**
Artificial Neural Network (ANN)	3.3776
Convolutional Neural Network (CNN)	13.3536
Recurrent Neural Network (RNN)	3.1278
Long Short-Term Neural Network (LSTM)	3.0146
Hybrid CNN-LSTM Model (Our Proposed Model)	**2.7378**

5.1.5 Comparison of the Results of the Hybrid CNN-LSTM with other Models

Model	**MAPE (%)**	**MSE**	**RMSE**	**MAE**
ANN	3.3776	3643.023	60.3574	40.531
CNN	13.3536	42392.877	205.8957	178.888
RNN	3.1278	2303.311	47.9928	38.840
LSTM	3.0146	2192.408	43.9843	37.094
Hybrid CNN-LSTM	**2.7378**	1849.049	38.0261	32.824

5.1.6 Comparison of results with results in literature

These are the results of the A2C-LNet and the PSO-A2C-LNet on the testing dataset compared to other models in scientific literature.

Model	**MAPE Results**
Hybrid CNN-LSTM Model (Our Proposed Model)	**2.737**
Integrated CNN and LSTM Network [23]	3.49
LSTM network considering attention mechanism[24]	2.26
ANN-IEAMCGM-R [25]	3.59
nonAda-GWN [26]	7.42
Ada-GWN [26]	6.83
GRU-CNN Hybrid Neural Network Model [27]	2.8839

6 Conclusion and Recommendation

6.1 Relevance of the Work

Our **Hybrid CNN-LSTM** Model has illuminated a groundbreaking path in the realm of energy forecasting by leveraging a hybrid CNN-LSTM network. This pioneering approach not only furthers the accuracy and efficiency of short-term load forecasting but also holds profound implications across a diverse spectrum of stakeholders, from power companies and individuals to governmental bodies and stock market specialists.

The significance of this work lies in its transformative impact on how various entities perceive and engage with demand prediction. For power companies, the application of advanced neural network architectures marks a pivotal stride toward optimized resource allocation, streamlined energy distribution, and enhanced grid stability. This, in turn, leads to reduced operational costs, minimized environmental footprints, and a heightened ability to accommodate the rising demand for sustainable energy practices.

Individuals stand to gain a newfound empowerment through this research's outcomes. Accurate load forecasting translates to optimized energy consumption patterns, enabling individuals to make informed decisions about their energy usage, thereby contributing to energy conservation and cost savings. The integration of reliable forecasts into governmental policy frameworks fosters resilient energy infrastructures, aiding in efficient energy allocation, crisis management, and disaster preparedness.

Furthermore, the influence extends to financial spheres, capturing the attention of stock market specialists who rely on precise predictions to inform investment decisions. Accurate load forecasting, offers a novel dimension for assessing energy-related stocks and portfolios. This research thus bridges the gap between the energy sector and financial markets, amplifying the strategic insights available to investors and ultimately influencing market dynamics.

6.2 Recommendations for Future Work

While the present research has pioneered a promising avenue for short-term load forecasting, several directions for future exploration beckon:

1. Enhanced Model Generalization: The integration of alternative attention mechanisms and exploration of novel meta-heuristic algorithms can potentially enhance the generalization capabilities of the proposed neural network architecture, accommodating diverse load patterns and unforeseen fluctuations.

2. Real-Time Implementation: Shifting the focus towards real-time implementation and deployment of the developed model could yield practical applications in demand response systems, enabling dynamic energy pricing and load management.

3. Incorporating External Factors: Integrating external factors such as weather conditions, social events, and economic indicators into the forecasting model could furnish a more comprehensive understanding of load variations, augmenting the accuracy of predictions.

4. Interdisciplinary Collaboration: Collaborative efforts between energy domain experts, data scientists, and economists can foster a holistic approach to load forecasting, embracing technical, behavioral, and economic facets.

5. Long-Term Load Forecasting: Extending the model's capabilities to long-term load forecasting opens avenues for infrastructure planning, energy market modeling, and sustainable energy transition strategies.

6.3 Recommendations

Our Model has shown promising results and the following are recommendations to various stakeholders and suggestions that will greatly the way things are done.

1. **Power Companies**:

   (a) **Operational Efficiency**: Implement the developed neural network architecture within your load forecasting system to improve short-term load predictions. This will lead to better resource planning, reduced energy wastage, and optimized power generation schedules.

(b) **Grid Stability**: Utilize the accurate load forecasts to enhance grid stability by efficiently allocating resources and managing demand fluctuations, thereby minimizing the risk of outages and disruptions.

(c) **Demand Response Programs**: Incorporate the forecasting model into demand response initiatives, encouraging consumers to adjust their energy consumption during peak load periods, which in turn helps to balance supply and demand.

2. **Individuals**:

   (a) **Energy Conservation**: Leverage the accurate load forecasts to adjust your energy consumption patterns during high-demand periods. By doing so, you can contribute to energy conservation efforts and potentially reduce your utility bills.

   (b) **Cost Savings**: Utilize the insights from load forecasting to plan energy-intensive activities during off-peak hours when energy prices are lower, leading to potential cost savings.

   (c) **Environmental Impact**: Align your energy consumption with periods of lower demand to indirectly reduce carbon emissions and support sustainable energy practices.

3. **Governmental Bodies**:

   (a) **Policy Formulation**: Integrate the research outcomes into energy policy frameworks to enhance energy infrastructure planning, crisis management, and disaster response.

   (b) **Renewable Integration**: Use accurate load forecasting to effectively integrate renewable energy sources into the grid, optimizing their contribution and minimizing reliance on fossil fuels.

   (c) **Data-Driven Decision Making**: Incorporate load forecasting insights into policy decisions regarding energy subsidies, tariffs, and incentive programs, creating a more efficient and responsive energy ecosystem.

4. **Financial Analysts and Investors**:

(a) **Investment Strategy**: Incorporate accurate load forecasts into investment strategies involving energy-related stocks, allowing for more informed decisions and potentially higher returns.

(b) **Risk Management**: Utilize load forecasting insights to assess potential risks and opportunities associated with energy market volatility, aiding in risk mitigation and portfolio diversification.

(c) **Market Analysis**: Leverage load forecasting data to analyze trends and correlations between energy demand and market fluctuations, gaining insights into market dynamics and potential investment trends.

5. **Researchers and Academics**:

   (a) **Model Advancement**: Build upon the research by exploring new attention mechanisms, optimization algorithms, or hybrid architectures to further refine short-term load forecasting accuracy.

   (b) **Interdisciplinary Collaboration**: Collaborate with professionals from diverse fields such as economics, sociology, and environmental science to create a comprehensive understanding of energy demand dynamics.

6. **Energy Technology Developers**:

   (a) **Model Integration**: Incorporate the proposed neural network architecture into energy management systems, smart appliances, and IoT devices to offer consumers more intelligent and efficient energy consumption solutions.

By tailoring their strategies and operations based on the recommendations above, these various entities can effectively harness the potential of our model to enhance their decision-making processes, optimize resource allocation, and contribute to a more sustainable energy future.

References

[1] D. O. Frederick and A. E. Selase, "The effect of electric power fluctuations on the profitability and competitiveness of smes: A study of smes within the accra business district of ghana," *Journal of Cryptology*, vol. 6, pp. 32–48, 2014.

[2] N. D. Rao and S. Pachauri, "Energy access and living standards: Some observations on recent trends," *Environmental Research Letters*, vol. 12, no. 2, p. 025 011, 2017.

[3] T. M. Letcher, *11-storing electrical energy, editor (s): Trevor m. letcher, managing global warming*, 2019.

[4] P. Jiang, F. Liu, and Y. Song, "A hybrid forecasting model based on date-framework strategy and improved feature selection technology for short-term load forecasting," *Energy*, vol. 119, pp. 694–709, 2017.

[5] Y. Chen, P. B. Luh, C. Guan, *et al.*, "Short-term load forecasting: Similar day-based wavelet neural networks," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 322–330, 2009.

[6] O. Ellabban, H. Abu-Rub, and F. Blaabjerg, "Renewable energy resources: Current status, future prospects and their enabling technology," *Renewable and sustainable energy reviews*, vol. 39, pp. 748–764, 2014.

[7] D. Ahmed, M. Ebeed, A. Ali, A. S. Alghamdi, and S. Kamel, "Multi-objective energy management of a micro-grid considering stochastic nature of load and renewable energy resources," *Electronics*, vol. 10, no. 4, p. 403, 2021.

[8] I. Ozer, S. B. Efe, and H. Ozbay, "A combined deep learning application for short term load forecasting," *Alexandria Engineering Journal*, vol. 60, no. 4, pp. 3807–3818, 2021, ISSN: 1110-0168. DOI: https://doi.org/10.1016/j.aej.2021.02.050. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S111001682100137X.

[9] K. Grolinger, M. Hayes, W. A. Higashino, A. L'Heureux, D. S. Allison, and M. A. Capretz, "Challenges for mapreduce in big data," in *2014 IEEE World Congress on Services*, 2014, pp. 182–189. DOI: 10.1109/SERVICES.2014.41.

[10] A. A. Author, B. B. Author, and C. Author, “Title of article,” *Title of Journal*, vol. 10, no. 2, pp. 49–53, 2005.

[11] S. S. Subbiah and J. Chinnappan, “Deep learning based short term load forecasting with hybrid feature selection,” *Electric Power Systems Research*, vol. 210, p. 108 065, 2022, ISSN: 0378-7796. DOI: https://doi.org/10.1016/j.epsr.2022.108065. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779622002875.

[12] W. Guo, L. Che, M. Shahidehpour, and X. Wan, “Machine-learning based methods in short-term load forecasting,” *The Electricity Journal*, vol. 34, no. 1, p. 106 884, 2021.

[13] C. Li, “A fuzzy theory-based machine learning method for workdays and weekends short-term load forecasting,” *Energy and Buildings*, vol. 245, p. 111 072, 2021.

[14] A. Forootani, M. Rastegar, and A. Sami, “Short-term individual residential load forecasting using an enhanced machine learning-based approach based on a feature engineering framework: A comparative study with deep learning methods,” *Electric Power Systems Research*, vol. 210, p. 108 119, 2022, ISSN: 0378-7796. DOI: https://doi.org/10.1016/j.epsr.2022.108119. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779622003431.

[15] A. Shaqour, T. Ono, A. Hagishima, and H. Farzaneh, “Electrical demand aggregation effects on the performance of deep learning-based short-term load forecasting of a residential building,” *Energy and AI*, vol. 8, p. 100 141, 2022, ISSN: 2666-5468. DOI: https://doi.org/10.1016/j.egyai.2022.100141. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666546822000052.

[16] A. Bellahsen and H. Dagdougui, “Aggregated short-term load forecasting for heterogeneous buildings using machine learning with peak estimation,” *Energy and buildings*, vol. 237, p. 110 742, 2021.

[17] M. Rahmoune and S. Chettih, “Forecasting of electricity demand by hybrid ann-pso under shadow of the covid-19 pandemic,” *European Journal of Electrical Engineering*, vol. 23, pp. 433–438, Dec. 2021. DOI: 10.18280/ejee.230602.

[18] S. Muzaffar and A. Afshari, “Short-term load forecasts using lstm networks,” *Energy Procedia*, vol. 158, pp. 2922–2927, 2019.

[19] N. M. M. Bendaoud and N. Farah, "Using deep learning for short-term load forecasting," *Neural computing and applications*, vol. 32, pp. 15 029–15 041, 2020.

[20] J. P. Holdren, "Population and the energy problem," *Population and Environment*, vol. 12, no. 3, pp. 231–255, Mar. 1991, ISSN: 1573-7810. DOI: 10.1007/BF01357916. [Online]. Available: https://doi.org/10.1007/BF01357916.

[21] A. Zaharia, M. Diaconeasa, L. Brad, G.-R. Lădaru, and C. Ioanas, "Factors influencing energy consumption in the context of sustainable development," *Sustainability*, vol. 11, p. 4147, Aug. 2019. DOI: 10.3390/su11154147.

[22] M. Han, A. Tan, and J. Zhong, "Application of particle swarm optimization combined with long and short-term memory networks for short-term load forecasting," *Journal of Physics: Conference Series*, vol. 2203, no. 1, p. 012 047, Feb. 2022. DOI: 10.1088/1742-6596/2203/1/012047. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/2203/1/012047.

[23] S. H. Rafi, Nahid-Al-Masood, S. R. Deeba, and E. Hossain, "A short-term load forecasting method using integrated cnn and lstm network," *IEEE Access*, vol. 9, pp. 32 436–32 448, 2021. DOI: 10.1109/ACCESS.2021.3060654.

[24] J. Lin, J. Ma, J. Zhu, and Y. Cui, "Short-term load forecasting based on lstm networks considering attention mechanism," *International Journal of Electrical Power Energy Systems*, vol. 137, p. 107 818, 2022, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2021.107818. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061521010346.

[25] P. Singh, P. Dwivedi, and V. Kant, "A hybrid method based on neural network and improved environmental adaptation method using controlled gaussian mutation with real parameter for short-term load forecasting," *Energy*, vol. 174, pp. 460–477, 2019, ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2019.02.141. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544219303408.

[26] W. Lin, D. Wu, and B. Boulet, "Spatial-temporal residential short-term load forecasting via graph neural networks," *IEEE Transactions on Smart Grid*, vol. 12, no. 6, pp. 5373–5384, 2021. DOI: 10.1109/TSG.2021.3093515.

[27] L. Wu, C. Kong, X. Hao, and W. Chen, “A Short-Term load forecasting method based on GRU-CNN hybrid neural network model,” *Mathematical Problems in Engineering*, vol. 2020, p. 1 428 104, Mar. 2020.