

Short Term Load Forecasting

Using Machine learning and Artificial Intelligence



Presented by:
Blessed Mukundi Mangena

Prepared for:
Professor Komla Folly
Dept. of Electrical and Electronics Engineering
University of Cape Town

Submitted to the Department of Electrical Engineering at the University of Cape Town
in partial fulfilment of the academic requirements for a Bachelor of Science degree in
Electrical and Computer Engineering

August 20, 2025

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.

Signature:.....

M. S. Tšoeu

Date:.....

Terms of Reference

The terms of reference page is an agreement between yourself and your supervisor outlining what is expected of you in your final year project. Please make sure that this is discussed and written at the beginning of your thesis project.

Acknowledgments

Make relevant acknowledgements to people who have helped you complete or conduct this work, including sponsors or research funders.

Abstract

- Open the **Project Report Template.tex** file and carefully follow the comments (starting with %).
- Process the file with **pdflatex**, using other processors may need you to change some features such as graphics types.
- Note the files included in the **Project Report Template.tex** (with the .tex extension excluded). You can open these files separately and modify their contents or create new ones.
- Contact the latex manual for more features in your document such as equations, subfigures, footnotes, subscripts & superscripts, special characters etc.
- I recommend using the **kile** latex IDE, as it is simple to use.

Contents

| | | |
|----------|---|----------|
| 1 | Introduction | 1 |
| 1.1 | Background to the study | 1 |
| 1.2 | Objectives of this study | 1 |
| 1.2.1 | Problems to be investigated | 1 |
| 1.2.2 | Purpose of the study | 1 |
| 1.3 | Scope and Limitations | 2 |
| 1.4 | Plan of development | 2 |
| 2 | Literature Review | 3 |
| 2.0.1 | Introduction | 3 |
| 2.1 | Statistical Models | 5 |
| 2.2 | Intelligent Models | 9 |
| 2.2.1 | Early Artificial Intelligence(AI) Models | 9 |
| 2.2.2 | Advanced Deep Learning Architecture for Enhanced STLf | 11 |
| 2.3 | Hybrid Models | 13 |

| | | |
|----------|---|-----------|
| 2.3.1 | Decomposition-Based Hybrid Models | 14 |
| 2.3.2 | Deep Learning Combination Hybrid Models | 14 |
| 2.4 | Data Pre-Processing Techniques | 16 |
| 3 | Methodology | 18 |
| 4 | Results | 19 |
| 4.1 | Simulation Results | 19 |
| 4.2 | Experimental Results | 19 |
| 5 | Discussion | 20 |
| 6 | Conclusions | 21 |
| 7 | Recommendations | 22 |
| A | Addenda | 28 |
| A.1 | Ethics Forms | 28 |

Chapter 1

Introduction

1.1 Background to the study

A very brief background to your area of research. Start off with a general introduction to the area and then narrow it down to your focus area. Used to set the scene .

1.2 Objectives of this study

1.2.1 Problems to be investigated

Description of the main questions to be investigated in this study.

1.2.2 Purpose of the study

Give the significance of investigating these problems. It must be obvious why you are doing this study and why it is relevant.

1.3 Scope and Limitations

Scope indicates to the reader what has and has not been included in the study. Limitations tell the reader what factors influenced the study such as sample size, time etc. It is not a section for excuses as to why your project may or may not have worked.

1.4 Plan of development

Here you tell the reader how your report has been organised and what is included in each chapter.

I recommend that you write this section last. You can then tailor it to your report.

Chapter 2

Literature Review

2.0.1 Introduction

An odd power surge phenomenon was recognized during the 2008 Olympics in the UK. At certain times in the evening there would be massive power surges at what seemed like random times in the evening. Though they seemed random to the utility companies they realized something rather funny was happening at these random times. Making a cup of tea is deeply ingrained in the english culture , after looking at the data they realized that the power surge was happening during times when a TV commercial would come on. Whenever a commercial came on TV multiple families would switch on their kettles to make a cup of tea. Though a single kettle may seem harmless , but millions of kettles switched on at the same time would cause a large effect on the load of the power grid. This phenomenon was called the '***Great British Kettle Surge***' [1].*you can also talk about the blackouts mentioned in [2]*

The effect of this great kettle surge can be very harmful to the grid if necessary steps are not taken to increase the grids capacity in times when we expect the power of the grid to increase suddenly. This is where the concept of load forecasting comes into play. Electric Load forecasting is the process of predicting how much electricity will be needed at a given time and how that demand will affect the utility grid [3]. To bring this matter close to home , South Africa has been struggling with provision of power to its population. This could be accredited to the lack of proper load forecasting in the previous years. A Growing population results in a growth in the demand of electricity. If the country does not build facilities to supply enough electricity for the load requirements then load-shedding becomes the only solution to protect the grid. In a perfect country the advice

from load forecasters in the early 2000's would have been taken into consideration and built more facilities to meet this demand. The above example shows the hand in hand relationship between load forecasts and their economic impacts. Large forecasting errors may lead to either excessively risky or excessively conservative scheduling, which can in turn result in undesirable economic penalties[4]. An addition in accuracy of 15 can save the UK 10 000 000 euros[5]. This means that there is a massive push towards finding the best load forecasting techniques that can be accurate and reliable.

Load forecasting is separated into 3 main categories which are Short , Medium and Long term Load Forecasting. the major differentiator between the three is the duration in which the forecasting is predicted for.

Long Term Load Forecasting(LTLF) considers periods that are more than a year. LTLF mainly considers factors such as demographic changes, economic growth and energy policy impacts [3]. This forecasting helps utilities think of what can be done to improve their systems to meet the increasing demand of the grid in the future. **Medium Term Load Forecasting (MTLF)** forecasts look at periods between a few months and a year. MTLF is important for demand side management , storage maintenance and scheduling of power [6].

Finally we have short term load forecasting(STLF) which looks at shorter time periods from hourly, daily all the way up to a week of load prediction. STLF is essential in daily operation performance , such as load flow and estimating how many power generators can be used in a particular day [7]. If there is efficient model for STLF , problems such as *british kettle surge* can easily be planned for ahead by ensuring more generators are operational when the demand for power rises. STLF can also ensure that the grid has a reliable continuous power flow during power shortages or outages[8].

In this study we would like to look at how we can improve STLF using machine learning and deep learning model approaches. There has been a clear identification in most of the current studies that Artificial Neural Networks offer a more accurate forecast than traditional statistical models. The major tradeoff between these two methods would be accuracy and computational power.

In this literature review we will look at the Statistical , Intelligent and Hybrid models that have been used for the purpose of prediction. We will also look at the different methods that have been used for data processing. This is because most models strive to get the highest accuracy value whilst , treating all data equally in the training process and not looking at the anomalies and different seasonal volatilities that represent themselves in

this data [9]

2.1 Statistical Models

Statistical methods have been the backbone of STLF before the prevalence of machine learning technologies. Different statistical models boast the ability to catch correlation in time series data. The methods have successfully been able to predict load with minimal error in the short and long term dimension. Though statistical methods are very old and have more cons than newer machine learning and hybrid models, they are still very relevant. [10] highlights that despite their disadvantages and low performances for time series data, statistical models still have a, "fast implementation in practice, a wide range of use and are well studied". In this section we will review some statistical models that have been used and are still in use and look at how they hold up with current advancements.

Regression Models

Regression algorithms are mathematical tools that establish statistical correlations between variables, providing an insight on how each of the data parameters are related to the output data [5]. It also allows using a lot of predictors and it will predict outcome when there are a lot of independent predictors. A regression model being fundamentally based on the principle of finding correlation between input parameters and the output [11], it works well only until our dataset has fluctuating data. [12] used Multiple Linear regression in their study which was successfully trained and optimized to give a 95% accuracy. Datasets for the STLF usually contain a lot of different data points and having an MLR allows us to consider each of the features to be added to the contribution.

[5] did a study to evaluate the most effective out of 24 statistical models for their functionality. Amongst them were regression trees, linear trees, exponential and quadratic trees just to name a few. These different regression models were tested under the same hardware conditions and the data pre-processing techniques and only 6 out of the 24 were shortlisted as the best performing, however the Rational Quadratic GPR and the Exponential GPR were the best rated regression models to use. These two models showed low error percentages. They benchmarked these models using an SVM and the regression models outperformed them. Though the regression model produces what may look like good results, they have massive limitations. Most regression models are "parametric

linear models and can capture only linear dependencies between the current sample and historical data”, [13]. This means that regression models struggle when the data has non-linear dependencies. The solution to the ineffectiveness of linear regression models could be using Support Vector Machines (SVM). This is because SVM’s help in classification problems where LR models fail to provide clear boundaries. A SVM is capable of finding the best separating boundary between different classes in the input data. **find source**. In the case of data that is not perfectly separable like the data we use for STLF SVM’s can find soft margins to classify data points. Even though we would be able to solve for these nonlinear aspects of the forecasts using SVM’s they become ineffective with very large datasets [11].

Exponential Smoothing

Exponential smoothing (ES) is a simple, low cost and adaptable statistical method that is used for time series forecasting. The concept behind ES is that the weights of the observed time series are exponentially decreased as observations come further in the past [14]. The most recent time points get the highest weights while the older time get progressively smaller weights [14]. There are different versions of exponential smoothing. The first one being Simple Exponential Smoothing, this is in its most basic form, and it puts focus on estimating the smoothed value of the series at a given time. The forecast that the SES will give for the next time step is determined by a smoothing constant [15]. [14] explains in his paper how ES methods are differentiated by their components and he simplified the components into three. The Level components which would focus on the smoothed average of the series, which would be the basis of all models. The second component is trend, which is the rate of change in the data and finally the Seasonality component which finds the recurring patterns in the data over a period. A combination of each of these components will end up giving us different ES models that specify on a particular component. The SES only uses the level component and this is a problem.

[16] using the component model also used the Holt-Winters model that is used for the seasonal time series data, and can be augmented with an auto-regressive error correction term. This model contains parameters for the level and seasonal components. This change separates them from the Simple ES because it can predict on both of these components rather than one. Finally they also tried the Double-Seasonal Holt-Winters-Taylor model, this model captures double seasonality found in higher frequency electricity demand series data. This makes it a good method for STLF.

ES models have several advantages. They are very simple to set up and have a quick

learning capability , they have a straightforward structure [13]. These models are also capable of capturing seasonal and trend variations. They are well suited for time series data that exhibits a strong trend of seasonal patterns which are very common in electric load data [14].

Despite the strengths ES models face limitations when looking at the complex characteristics of electric load data. These models are essentially based on linear analysis and struggle to capture the inherent non-linearity and high volatility present in electricity load time series, which are influenced by numerous factors like weather, holidays, and user habits [13]. Their performance can deteriorate with highly irregular and random time series data [17] , essentially saying there are sensitive to noise and irregularities. Finally ES models generally tend to produce less accurate results than the "black box" methods used by utility companies [18].

Auto Regressive Integrated Moving Average

The Autoregressive Integrated Moving Average (ARIMA) model is a widely recognised statistical approach employed for time series forecasting, particularly in Short-Term Load Forecasting (STLF) within power systems. It is considered as a linear model that combines autoregressive (AR) , difference (I) and moving average (MA) components [19]. Its valued as a simple and efficient model and is primarily excellent at capturing linear relations and periodic patterns in time series data [14].

The ARIMA framework is often referred as the Box-Jenkins model , its systematic and practical and involves 3 iterative steps.

Step 1 : Model Identification The initial stage involves analysing time series data to determine the appropriate orders (p, d , q) for the AR , I and MA components. The AR component signifies that the current value of the time series (Y_t) is expressed linearly in terms of its 'p' previous values and a random noise component [20]. The MA component denotes that the current value of Y_t is a linear combination of white noise error terms (e_t) from previous time steps. The integrated (I) component (represented by 'd') signifies that the time series has been differenced 'd' times to achieve stationarity [15].

Step 2 : Parameter Estimation Once the model structure (p, d, q) is identified, the parameters for the AR and MA components are estimated. This is often achieved by maximising the log-likelihood function [14].

Step 3 : Model Diagnosis The final stage involves assessing the fit of the model by checking if the residuals are uncorrelated (i.e., behave like white noise). If the residuals are not uncorrelated, the model needs to be refined, requiring a return to the identification or estimation steps [14].

ARIMA has been extended in multiple studies to enhance its capabilities . The Seasonal Auto-Regressive Integrated Moving Average (SARIMA) , is especially designed to handle seasonality in the data such as looking at daily patterns recognized in the data [21]. Eventually a SARIMA model can effectively eliminate the influence of periodicity in the prediction and it has shown great results in the past tests [22].

In a study conducted by Y. wang et al [22] where they were looking at the performance of three residual modification for improving SARIMA. In this study they implemented an optimized Fourier residual modification and this residual improved the outputs accuracy more than the normal SARIMA. The SARIMA/ARIMA are statistical models and carry the same disadvantages. Even though it has higher accuracy than ES models it is still inferior to ANN and SVM models [23]. In the study by Jiang et al [23] they found that ARIMA less accurate, but notably more computationally efficient (11.25 seconds for ARIMA versus 683.62 seconds for ANN and 1412.7 seconds for GA-SVM). Hybrid ARIMA models would outperform standard ARIMA models however they would still perform inferior to machine learning models.

Conclusion for statistical methods

The statistical models offer a low cost of computation and easy implementation of LF. However since they fail to effectively capture nonlinear characteristics of load data they fail to satisfy the accuracy and stability that is required by the grid management in the modern day [24].

2.2 Intelligent Models

The emergence of intelligent methods in STLF has significantly advanced the field , moving beyond traditional statistical models to embrace computational techniques that can better handle the complex and non-linear nature of electricity demand [25].

At the dawn of Artificial intelligence were foundational models and tools that were used and implemented to held STLF accuracy. The early AI applications marked a crucial shift from purely statistical approaches that were limited by computing capabilities and often struggled with the non-linear features of time series data [26]. These models aimed to address these limitations by leveraging their ability to learn patterns from complex data but they did not do so well due to no linearity.

We will start by looking at very fundamental models that have been ued for prediction and how they have impacted the industry , we will expand further into looking at the more advanced models that are currently being implemented and tested in modern day technology.

2.2.1 Early Artificial Intelligence(AI) Models

Support Vector Machines (SVM) SVM's and their variants such as the Support Vector Regression(SVR) and Least Squares Support Vector Machine (LSSVM) were among the first prominent techniques used in STLF [26]. The basic idea of an SVM for regression , is to ,”map the data x into a high-dimensional feature space via a nonlinear mapping and to perform a linear regression in this feature space” [27]. This basically simplifies to an SVM being used in a classification task to choose a boundary that will maximize the margin that classifies an element in the dataset.

SVM's are renowned for their kernel trick that effectively handles non-linear input spaces and provides proficient prediction models for regression problems like load forecasting[28]. They overcome overfitting issues through kernel methods and regularization techniques [28] , however a drawback is identified in their ineffectiveness or intolerable long training times when dealing with large datasets [29]. Despite this SVM , has shown superiority to traditional statistical methods when it comes to load forecasting [5].

Fuzzy Logic methods also emerged as early AI tools for STLTF. Fuzzy logic is an extension of boolean logic however instead of having true or false (0 or 2) , fuzzy logic allows variability of the truth ranging between 0 and 1. For example a pot can be 0.7 for "hot" and 0.3 for "warm". Fuzzy logic approaches used fuzzy rules to integrate historical load data with time and day characteristics to determine probable load curves [30].

Artificial Neural Networks represented a significant leap forward due to their ability to model complex non-linear relationships between loads and related factors [29]. Mimicking the human brain's information processing , ANN's can approximate non-linear functions with high fidelity and accuracy , making them well suited for load forecasting [25]. ANN's gained their popularity due to their flexibility and robustness in handling diverse input-output projections and architectures. The earliest forms of ANN's for short term load forecasting showed up in the early 90's in the form of Multi Layer Perceptrons (MLP) [25] but have since grown into more advanced models.

Challenges and Drawbacks of Early AI Models

Despite the initial promise , standard ANNs presented key problems that limited their efficiency for time series tasks like STLTF. These Methods had *inefficiencies of Backpropagation*. The training of ANNs, especially those relying on backpropagation (BP) algorithms, suffered from issues such as slow convergence rates and high computational costs [29]. ANNs were also prone to getting stuck in local optima and exhibiting overfitting, which could lead to poor generalization on new, unseen data [31]. The final results of ANN models could also be dependent on the initial random weights and thresholds , contributing to forecast instability [25].

The early models also had a *Lack of Temporal Memory*. A fundamental limitation of standard ANNs for time series forecasting tasks was always their inability to learn from the sequential nature of the load data. Typically these models would treat each of the time steps individually when the time points are largely correlated. This would result in failure to memorize past information or account for long-term dependencies inherent in time varying electricity consumption patterns [31]. This meant that they struggled to capture the influence of previous timesteps on current or future load values , hindering accurate predictions for dynamic systems.

2.2.2 Advanced Deep Learning Architecture for Enhanced STLF

Deep learning techniques, characterized by a significantly higher number of layers, offered a powerful solution to these limitations, enabling models to deal with complicated non-linear patterns and learn complete probability distributions with temporal memory from vast datasets [13]. The vast data that is generated daily by smart grids and IoT devices makes deep learning models ideal due to their scalability. Deep Learning methods often offer better performance than more conventional machine learning methods [32].

Recurrent Neural Networks and Long Short Term Memory

Recurrent Neural Networks (RNNs) were introduced as a direct solution to the temporal memory problem in neural networks. Unlike the feed-forward networks, RNNs establish weight connections between layers over time, allowing them to reflect sequential information and possess a "memory ability" [26]. This made them suitable for time series data, where the current outputs depend on the previous inputs and hidden states. This is because RNNs introduce a loop so that the network can remember information from previous steps.

However, original RNNs still faced the vanishing gradient problem, where the ability of later time nodes to perceive previous ones decreased as the network deepened, limiting their performance with long time sequences [26].

To address this, the Long Short-Term Memory (LSTM) network architecture was proposed, enhancing RNNs with recurrent gates (input, output, and forget gates) to solve the vanishing gradient problem and effectively deal with long-term dependencies [?]. LSTM networks contain memory cells that store information over random time intervals, and gates that trace the flow of input and output data from the cell, allowing them to determine what information to discard, store, or output selectively [33]. Studies have consistently demonstrated LSTM's effectiveness, showing lower errors and higher accuracy in STLF compared to traditional ANN approaches. They are particularly capable of handling more complex time series load data with long-term dependencies [30]. LSTM has shown superior performance for longer forecast horizons compared to ARMA models [13] and this is because of their temporal capabilities.

Bidirectional LSTM (Bi-LSTM)

add an image for LSTM and BI-LSTM to show their difference

Bidirectional LSTM (Bi-LSTM) networks significantly improve upon the standard LSTM by processing data in both forward and backward directions [31]. This architecture consists of two independent LSTM layers that run in opposite directions, both connected to the same output layer [34]. This allows the model to leverage both past context (from the forward pass) and future context (from the backward pass) for more accurate and robust predictions. The bidirectional movement and interconnected structure help eliminate problems such as missing data and overfitting in the training phase [34]. Research has shown Bi-LSTM to achieve superior performance, often with significantly lower Mean Absolute Percentage Error (MAPE) values, compared to unidirectional LSTMs and other methods [32].

Deep Belief Network (DBN)

Deep Belief Networks (DBNs) emerged as a crucial solution to address some of the challenges associated with backpropagation, particularly the difficulty of finding optimal initial parameters and the problem of local optima [35]. DBNs are generative probabilistic models composed of stacked Restricted Boltzmann Machines (RBMs) and a classifier. DBNs combine deep learning with feature learning, giving them powerful fitting capabilities to quickly analyze large amounts of data [35]. The multiple layers in DBNs help make them better in handling multifactor relationships in comparison to single layer networks. Their greedy, layer-wise unsupervised pre-training mechanism helps in identifying better initial parameters, which are then fine-tuned through supervised learning [35]. This pre-training process allows DBNs to learn patterns progressively and overcome the disadvantage of being easily trapped in local optima due to random weight initialization. By converting continuous input features into binomial distribution features (e.g., using Gauss-Bernoulli RBMs), DBNs can improve their effectiveness in processing real-valued data [35]. This approach leads to faster convergence and improved prediction performance in various fields, including load forecasting. This makes them ideal for uncovering electricity usage patterns and are considered ideal due to their scalability [36].

Other Advanced Models

Gated Recurrent Unit (GRU) neural networks were proposed as a simpler alternative to LSTM, combining the input and forget gates into a single "update gate" [26]. GRUs can achieve similar accuracy to LSTMs with less computational cost and faster convergence [37]. They are effective at extracting temporal features and preserve important features while using fewer parameters [37].

Convolutional Neural Networks (CNN) have been frequently used in load prediction due to their excellent ability to capture the trend of load data. CNNs are particularly effective for feature extraction and dimension reduction of original data [37]. CNNs have a lack of temporal memory that put them at a disadvantage in forecasting tasks separately that require memory. They are often integrated with other models like LSTMs to leverage both spatial feature extraction and temporal modeling capabilities [30]. A method by Shafiul Hasan Rafi proposes an integrated CNN and LSTM network for STLTF in the Bangladesh power system, reporting higher precision and accuracy than existing approaches like LSTM, Radial Basis Function Network (RBFN), and XGBoost [30].

2.3 Hybrid Models

Hybrid models have emerged as a prominent approach in Short-Term Load Forecasting (STLF) to address the inherent complexities of electricity demand prediction, such as its non-linear and non-stationary characteristics [38]. These models combine the strengths of various algorithms and techniques to enhance forecasting accuracy and reliability, overcoming the limitations often encountered by single, standalone models.

The fundamental motivation is that no single forecasting technique is superior in all cases [24]. By combining methods, hybrid models can effectively handle the non-linear and multivariate characteristics of load data, extract complex patterns, and improve prediction accuracy and stability [24]. Conceptually, they operate by breaking down the forecasting problem, applying specialized techniques to different aspects (e.g., feature extraction, temporal learning, error correction), and then integrating the results. This often involves preprocessing data, applying deep learning for feature learning and sequence modeling, and optimizing parameters [35].

2.3.1 Decomposition-Based Hybrid Models

A common approach involves decomposing the original, complex load data into simpler, more stable components using techniques like Variational Mode Decomposition (VMD) [33], Empirical Mode Decomposition (EMD), or complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) [37]. These methods will solve the problem of analyzing time series data directly and solve this by splitting the data into intrinsic components that are easier for machine learning models to learn from. Each component is then predicted using an appropriate model like an LSTM for high-frequency components, Multiple Linear Regression (MLR) for low-frequency components [39] and the results are aggregated for the final forecast. This pre-processing step significantly reduces volatility and improves prediction accuracy [31].

2.3.2 Deep Learning Combination Hybrid Models

CNN-LSTM/GRU

Hybrid models like CNN-LSTM and CNN-GRU are frequently used. The CNN module excels at extracting local features and patterns from time series data, while the LSTM or GRU module is proficient in capturing long-term temporal dependencies [40]. The combination of such models help get the benefits of both in a prediction task.

The CNN module is typically used to extract local features and patterns from load data [41]. This is achieved through pooling and convolution layers which process two-dimensional data and flatten it into a one-dimensional feature vector. This feature vector is then fed as input to the LSTM or GRU layers, which are good at capturing long-term dependencies and sequential information in time series data [40]. LSTM mechanism allow for the retaining of long-term information and address the vanishing gradient problem [40]. The GRU would combine with the CNN in the same way as an LSTM as they have a similar structure just simpler and fewer parameters, making it faster [26].

Such a combination will improve the forecast accuracy, reduce prediction errors and outperform single models. In the tests done by Rafi et al [30], they found that CNN-LSTM models have a higher precision and accuracy in short-term load forecasting compared to standalone LSTM, Radial Basis Function Network (RBFN), and Extreme Gradient Boosting (XGBoost) approaches. A CNN-LSTM model provided 173.76 MW less Mean Absolute Error (MAE), 330.2 MW less Root Mean Squared Error (RMSE), and 3.07%

less Mean Absolute Percentage Error (MAPE) on average than a standalone LSTM network. In another study done by Danish et al [42] they found that the GRU-CNN model outperformed a GRU, CNN and a Back Propagation Neural Network (BPNN) and this was due to its capability to learn from both time sequence data and spatiotemporal data.

DBN-RNN/Bi-RNN

Deep Belief Networks (DBNs) are integrated with Recurrent Neural Networks (RNNs) or Bidirectional RNNs (Bi-RNNs). This hybrid model typically leverages the DBNs capability for unsupervised pre-training and optimal parameter seeking with the RNN's strength in processing sequential and time-dependent data [43].

A DBN being a generative and probabilistic model composed of multiple layers of RBM. These RBM combine the traditional neural networks with energy and probabilistic models [38]. The RBM is crucial for initialising parameters and this pretraining helps overcome issues like local minimisation and slow convergence seen in some other neural network [44]. On the other side the RNN is specifically designed to process sequence data due to their internal loop structure that allows information to persist across time steps [26] and this makes them highly effective for STLTF where mining large quantities of temporal data is essential. A Bi-RNN is just a structural improvement that will process the input simultaneously in two opposite directions (forward and backward) [43]. This bidirectional structure gives the RNN a complete past and future context information for each point in the input layer, enhancing the validity of the forecasting model [43].

In the test done by Tang et al [43] they used the DBN to perform the unsupervised greedy pre-training layer by layer to obtain initial weight parameters, making it easier to approach optimal values. The pretraining phase helped in handling parameters initialization problem in a large dataset. After this pre-training the parameters are supervisedly adjusted through the Bi-RNN, which then propagates learned time information in both directions to derive the final forecasted value. In the same study they also incorporated data preprocessing which are Bisecting K-Means Algorithm for clustering and Ensemble Empirical Mode Decomposition (EEMD) for decomposing load data into intrinsic mode functions (IMFs), is employed to reduce noise and enhance feature selection using methods like Pearson correlation coefficient. This data preprocessing played a part in the final performance of this hybrid model. The results of this study show that DBN with Bi-RNN reduces the MAPE to 1.85% from 2% which is a move towards the right direction. The DBN-Bi-RNN models have been shown to outperform unidirectional LSTM, SVR, and BPNN [43].

The DBN-RNN/Bi-RNN hybrid models provide a robust and accurate solution for STLTF by combining DBN's parameter optimisation and feature learning capabilities with RNN/Bi-RNN's strength in handling temporal dependencies and contextual information. They often outperform traditional and even some advanced deep learning models in terms of forecasting accuracy and computational efficiency [38].

2.4 Data Pre-Processing Techniques

2.4. DATA PRE-PROCESSING TECHNIQUES

- [31] talks about a data preprocessing and CNN-BI-LSTM
- paper on **(HI-CEEMDAN-Q-TEG)** [37] important
- this paper looks at an alternate **data preprocessing** method, it shows an improvement after adding sampling [9].
- **Data Processing** focuses on using sliding fuzzy granulation for data processing [24]. Offers the benefits for using statistical models. Li also mentions how hybrid models have been able to improve the forecast capabilities of the models .
- **data preprocessing** [33] it uses a VMD method. it is also connected to the [37]
- **Data preprocessing** [17] using the LMD processing technique
- **Data preprocossessing** [15] 3.2. Weather Classification Weather status "is an indispensable pre-processing step, especially for short-term PVPF (ST-PVPF)." PV output is "most strongly correlated with solar spectral irradiance," which depends on meteorological factors (aerosol distribution, wind speed and direction, humidity, and cloud cover).

Benefits: Integrating weather classification improves model robustness and forecast precision. Challenges: Insufficient training datasets for all weather types, particularly extreme conditions. Generative Adversarial Networks (GANs) are recommended for augmenting these datasets. Methodology: Most studies categorise weather into fewer than four types, though some reclassify up to 10. "Finer the weather classification; better the prediction results." Cloud Motion Study: Crucial for intra-hour forecasts. Ground-based sky images offer higher resolution than satellite imagery for regional cloud formations. Techniques like "image phase shift invariance (IPSI) based CMDV calculation married to Fourier phase correlation theory (FPCT)" are used.

- [13] showed how much superior an LSTM is to the traditional ARMA model
- [2] **CEEDMAN**

Chapter 3

Methodology

This is what I did to test and confirm my hypothesis.

You may want to split this chapter into sub chapters depending on your design. I suggest you change the title to something more specific to your project.

This is where you describe your design process in detail, from component/device selection to actual design implementation, to how you tested your system. Remember detail is important in technical writing. Do not just write I used a computer give the computer specifications or the oscilloscopes part number. Describe the system in enough detail so that someone else can replicate your design as well as your testing methodology.

If you use or design code for your system, represent it as flow diagrams in text.

IMPORTANT: Include a motivation for your selection of an appropriate technique, or engineering or IT tool to solve your project problem. Discuss any limitations if appropriate.

Chapter 4

Results

These are the results I found from my investigation.

Present your results in a suitable format using tables and graphs where necessary. Remember to refer to them in text and caption them properly.

4.1 Simulation Results

4.2 Experimental Results

Chapter 5

Discussion

Here is what the results mean and how they tie to existing literature...

Discuss the relevance of your results and how they fit into the theoretical work you described in your literature review.

Chapter 6

Conclusions

These are the conclusions from the investigation and how the investigation changes things in this field or contributes to current knowledge...

Draw suitable and intelligent conclusions from your results and subsequent discussion.

Chapter 7

Recommendations

Make sensible recommendations for further work.

Bibliography

- [1] R. BURDETT-GARDINER, “The great british kettle surge.” <https://www.renewableenergyhub.co.uk/blog/the-great-british-kettle-surge>, 2023. Accessed: 2025-08-05.
- [2] K. Li, W. Huang, G. Hu, and J. Li, “Ultra-short term power load forecasting based on ceemdan-se and lstm neural network,” *Energy and Buildings*, vol. 279, p. 112666, 2023.
- [3] A. McGrath, “What is load forecasting?.” <https://www.ibm.com/think/topics/load-forecasting>, 2024. Accessed: 2025-08-05.
- [4] S. Tzafestas and E. Tzafestas, “Computational intelligence techniques for short-term electric load forecasting,” *Journal of Intelligent and Robotic Systems*, vol. 31, no. 1, pp. 7–68, 2001.
- [5] S. Gochhait and D. Sharma, “Regression model-based short-term load forecasting for load dispatch centre,” *Journal of Applied Engineering and Technological Science (JAETS)*, vol. 4, no. 2, pp. 693–710, 2023.
- [6] L. Han, Y. Peng, Y. Li, B. Yong, Q. Zhou, and L. Shu, “Enhanced deep networks for short-term and medium-term load forecasting,” *IEEE Access*, vol. 7, pp. 4045–4055, 2018.
- [7] J. Shohan, M. Faruque, and S. Foo, “Forecasting of electric load using a hybrid lstm–neural prophet model. *energies* 2022, 15, 2158,” 2022.
- [8] C. Tarmanini, N. Sarma, C. Gezezin, and O. Ozgonenel, “Short term load forecasting based on arima and ann approaches,” *Energy Reports*, vol. 9, pp. 550–557, 2023.
- [9] C. Wang, Y. Zhou, Q. Wen, and Y. Wang, “Improving load forecasting performance via sample reweighting,” *IEEE Transactions on Smart Grid*, vol. 14, no. 4, pp. 3317–3320, 2023.

- [10] A. Rusina, T. Osgonbaatar, and P. Matrenin, “Short-term load forecasting using statistical methods for the central power system of mongolia,” *2022 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, pp. 2030–2035, 2022.
- [11] B. S. Vardhan, M. Khedkar, I. Srivastava, P. Thakre, and N. D. Bokde, “A comparative analysis of hyperparameter tuned stochastic short term load forecasting for power system operator,” *Energies*, vol. 16, no. 3, p. 1243, 2023.
- [12] B. Dhaval and A. Deshpande, “Short-term load forecasting with using multiple linear regression,” *International Journal of Electrical and Computer Engineering*, vol. 10, pp. 3911–3917, 2020.
- [13] O. T. Tshipata, D. T. Kazumba, P. S. Nzakuna, V. Paciello, and A. K. Lusala, “Multi-horizon short-term electrical load forecasting: a comparative analysis of statistical models and deep neural networks,” in *2024 IEEE International Symposium on Measurements & Networking (M&N)*, pp. 1–6, IEEE, 2024.
- [14] P. Ramos, N. Santos, and R. Rebelo, “Performance of state space and arima models for consumer retail sales forecasting,” *Robotics and computer-integrated manufacturing*, vol. 34, pp. 151–163, 2015.
- [15] R. Ahmed, V. Sreeram, Y. Mishra, and M. Arif, “A review and evaluation of the state-of-the-art in pv solar power forecasting: Techniques and optimization,” *Renewable and sustainable energy reviews*, vol. 124, p. 109792, 2020.
- [16] J. F. Rendon-Sanchez and L. M. de Menezes, “Structural combination of seasonal exponential smoothing forecasts applied to load forecasting,” *European Journal of Operational Research*, vol. 275, no. 3, pp. 916–924, 2019.
- [17] R. Wang, J. Wang, and Y. Xu, “A novel combined model based on hybrid optimization algorithm for electrical load forecasting,” *Applied Soft Computing*, vol. 82, p. 105548, 2019.
- [18] H. Takeda, Y. Tamura, and S. Sato, “Using the ensemble kalman filter for electricity load forecasting and analysis,” *Energy*, vol. 104, pp. 184–198, 2016.
- [19] V. Revathi, K. Prashant, A. Singla, B. Boddu, A. A. Hameed, S. Kalyani, and K. Pandey, “Short-term load forecasting for virtual power plants using time series analysis and open energy data,” in *2025 International Conference on Cognitive Computing in Engineering, Communications, Sciences and Biomedical Health Informatics (IC3ECSBHI)*, pp. 741–746, IEEE, 2025.

- [20] S. Dai Haleema, “Short-term load forecasting using statistical methods: A case study on load data,” *Int. J. Eng. Res. Technol.*, vol. 9, pp. 516–520, 2020.
- [21] M. Abbas, Y. Che, S. Maqsood, M. Z. Yousaf, M. Abdullah, W. Khan, S. Khalid, M. Bajaj, and M. Shabaz, “Self-adaptive evolutionary neural networks for high-precision short-term electric load forecasting,” *Scientific Reports*, vol. 15, no. 1, p. 21674, 2025.
- [22] Y. Wang, J. Wang, G. Zhao, and Y. Dong, “Application of residual modification approach in seasonal arima for electricity demand forecasting: A case study of china,” *Energy Policy*, vol. 48, pp. 284–294, 2012.
- [23] H. Jiang, Y. Zhang, E. Muljadi, J. J. Zhang, and D. W. Gao, “A short-term and high-resolution distribution system load forecasting approach using support vector regression with hybrid parameters optimization,” *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3341–3350, 2016.
- [24] S. Li, J. Wang, H. Zhang, and Y. Liang, “Short-term load forecasting system based on sliding fuzzy granulation and equilibrium optimizer,” *Applied Intelligence*, vol. 53, no. 19, pp. 21606–21640, 2023.
- [25] A. Arvanitidis, D. Bargiotas, A. Daskalopulu, V. Laitzos, and L. Tsoukalas, “Enhanced short-term load forecasting using artificial neural networks. energies 2021, 14, 7788,” 2021.
- [26] Y. Wang, M. Liu, Z. Bao, and S. Zhang, “Short-term load forecasting with multi-source data using gated recurrent unit neural networks,” *Energies*, vol. 11, no. 5, p. 1138, 2018.
- [27] M. Mohandes, “Support vector machines for short-term electrical load forecasting,” *International Journal of Energy Research*, vol. 26, no. 4, pp. 335–345, 2002.
- [28] M. Hussien, W. Yehia, and A. B. El-Sisi, “A comparative study of machine learning algorithms for short-term electrical load forecasting,” *IJCI. International Journal of Computers and Information*, vol. 8, no. 2, pp. 32–37, 2021.
- [29] X. Dong, L. Qian, and L. Huang, “Short-term load forecasting in smart grid: A combined cnn and k-means clustering approach,” in *2017 IEEE international conference on big data and smart computing (BigComp)*, pp. 119–125, IEEE, 2017.
- [30] S. H. Rafi, S. R. Deeba, E. Hossain, *et al.*, “A short-term load forecasting method using integrated cnn and lstm network,” *IEEE access*, vol. 9, pp. 32436–32448, 2021.

- [31] N. Wang and Z. Li, “Short term power load forecasting based on bes-vmd and cnn-bi-lstm method with error correction,” *Frontiers in Energy Research*, vol. 10, p. 1076529, 2023.
- [32] B. Ibrahim, L. Rabelo, E. Gutierrez-Franco, and N. Clavijo-Buritica, “Machine learning for short-term load forecasting in smart grids,” *Energies*, vol. 15, no. 21, p. 8079, 2022.
- [33] F. He, J. Zhou, Z.-k. Feng, G. Liu, and Y. Yang, “A hybrid short-term load forecasting model based on variational mode decomposition and long short-term memory networks considering relevant factors with bayesian optimization algorithm,” *Applied energy*, vol. 237, pp. 103–116, 2019.
- [34] A. Moradzadeh, H. Moayyed, S. Zakeri, B. Mohammadi-Ivatloo, and A. P. Aguiar, “Deep learning-assisted short-term load forecasting for sustainable management of energy in microgrid,” *Inventions*, vol. 6, no. 1, p. 15, 2021.
- [35] X. Kong, C. Li, F. Zheng, and C. Wang, “Improved deep belief network for short-term load forecasting considering demand-side management,” *IEEE transactions on power systems*, vol. 35, no. 2, pp. 1531–1538, 2019.
- [36] P. Boopathy, M. Liyanage, N. Deepa, M. Velavali, S. Reddy, P. K. R. Maddikunta, N. Khare, T. R. Gadekallu, W.-J. Hwang, and Q.-V. Pham, “Deep learning for intelligent demand response and smart grids: A comprehensive survey,” *Computer science review*, vol. 51, p. 100617, 2024.
- [37] J. Wang, H. Liu, G. Zheng, Y. Li, and S. Yin, “Short-term load forecasting based on outlier correction, decomposition, and ensemble reinforcement learning,” *Energies*, vol. 16, no. 11, p. 4401, 2023.
- [38] Y. Dong, Z. Dong, T. Zhao, Z. Li, and Z. Ding, “Short term load forecasting with markovian switching distributed deep belief networks,” *International Journal of Electrical Power & Energy Systems*, vol. 130, p. 106942, 2021.
- [39] W. Huang, Q. Song, and Y. Huang, “Two-stage short-term power load forecasting based on ssa-vmd and feature selection,” *Applied Sciences*, vol. 13, no. 11, p. 6845, 2023.
- [40] J. Zhu, J. Yang, X. Cui, M. Peng, and X. Liang, “A novel adaptive adjustment kolmogorov-arnold network for heat load prediction in district heating systems,” *Applied Thermal Engineering*, p. 126552, 2025.

- [41] 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, no. 1, p. 1428104, 2020.
- [42] M. U. Danish and K. Grolinger, “Kolmogorov–arnold recurrent network for short term load forecasting across diverse consumers,” *Energy Reports*, vol. 13, pp. 713–727, 2025.
- [43] X. Tang, Y. Dai, Q. Liu, X. Dang, and J. Xu, “Application of bidirectional recurrent neural network combined with deep belief network in short-term load forecasting,” *IEEE Access*, vol. 7, pp. 160660–160670, 2019.
- [44] Y. Gao, Y. Hang, and M. Yang, “A cooling load prediction method using improved ceemdan and markov chains correction,” *Journal of Building Engineering*, vol. 42, p. 103041, 2021.

Appendix A

Addenda

A.1 Ethics Forms