

https://dspace.org/

Investigating Design Parameters for Practical Load Forecasting of Grid-Interactive Buildings Using LSTM

Simani, Kyppy Ngaira

2024
University of the Witwatersrand, Johannesburg
https://hdl.handle.net/10539/45444
Simani, Kyppy Ngaira . (2024). Investigating Design Parameters for Practical Load Forecasting of Grid-Interactive Buildings Using LSTM [Masters dissertation, University of the Witwatersrand, Johannesburg]. WIReDSpace.

Downloaded from a DSpace repository. https://dspace.org/

University of the Witwatersrand

Masters Dissertation

Investigating Design Parameters for Practical Load Forecasting of Grid-Interactive Buildings Using LSTM

Author:
Kyppy Simani

A Dissertation submitted in fulfillment of the requirements for the degree of Master of Science

in the

Power and Energy Research Group
School of Electrical and Information Engineering

September 2024



UNIVERSITY OF THE
WITWATERSRAND,
JOHANNESBURG

Declaration

I, **Kyppy Simani**, declare that this Dissertation titled, "**Investigating Design Parameters for Practical Load Forecasting of Grid-Interactive Buildings Using LSTM**" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this dissertation is entirely my own work.
- I have acknowledged all main sources of help.

Signed: 

Date: 4th September, 2024

Abstract

The dissertation presented contributes to research in residential load forecasting using machine-learning (ML) models to optimise energy management within grid-interactive efficient buildings (GEB) and addresses the challenge of implementing forecasting models for practical applications.

The use of accurate load forecasting ML models has been shown to extend and improve the field of energy management towards the resource coordination of GEB. Developing these models is a time-, data- and compute-intensive process that requires selection and tuning of various design parameters. In practice however this has costs and limitations: collection and storage of real-time data, measurement device costs and finite compute power. To simplify model development and reduce these costs, this dissertation investigates to what extent the relationships between model parameters and performance can inform ML design decisions.

In the research presented, 3 experiments are performed to investigate the relationships between training data size, prediction horizon length and data resolution and their impact on the predictive model fidelity. Several prediction models are developed using various parameter settings. The accuracy and compute intensity are measured to evaluate the influence of each parameter on the model performance. A dataset of 3 years and 11 months of residential power consumption measurements at a one-minute resolution is used as the load profile for this investigation. For this load profile it is found that increasing training data size increases compute time linearly, with an exponential decay in model prediction error. This results in a maximum, resource-efficient training data size of 450 days. In addition, the effective prediction (i.e prediction models with R^2 scores greater than 0) horizon length is found to be 3 hours. At this length the load profile is suited for short-term load forecast applications such as distributed energy resource (DER) and real-time pricing (RTP) management. Furthermore, models trained on low-resolution data (up to 30 minutes) can achieve comparable performance to higher-resolution models with at least 3 additional months of training data. The findings of this investigation therefore represent a contribution towards the development of a ML design tool for more efficient design parameter tuning considering practical load forecasting conditions. This shows potential to enable more efficient implementations of load forecasting systems for GEB environments.

I dedicate this to my Mother

Ms Doris Mwikali Nzuki, who always made sure I had what I needed to succeed, even when I was afraid to.

I dedicate this to my Father

Mr Peter Ldhituachi Simani, who showed me that hard work pays and how to laugh and shrug it off when it doesn't.

Acknowledgements

I would like to thank, with the utmost respect and appreciation, my supervisors Dr. Yu-chieh Yen and Dr. Yuval Genga for the patience, guidance and support they have provided me throughout the process of completing this dissertation. Without their incredible enthusiasm, patience, and knowledge I would not have been able to complete this journery and produce this work.

I would also like to thank all of my friends for their help, hugs and laughter, and for always reminding me how to live outside of engineering.

Finally I want to extend a big thanks to my family. Their unwavering support gave me the courage and means to take a chance on myself and pursue this oppurtunity. An extra-special thanks to my partner Kelsey Wishnuff whose kindness, patience, encouragement and love gave me a place to come home to and made completion of this work possible.

Contents

Declaration i

Abstract ii

Dedication iii

Acknowledgements iv

Contents v

List of Figures viii

List of Tables x

Abbreviations xi

Publications xiii

1 Introduction 1
- 1.1 Problem Statement . . . 2
- 1.2 Approach . . . 2
- 1.3 Research Question and Objectives . . . 3

2 Background and Literature Review 5
- 2.1 Residential Energy Demand . . . 5
  - 2.1.1 Urbanisation . . . 6
- 2.2 Residential Energy Management . . . 7
  - 2.2.1 Demand-side Management . . . 7
  - 2.2.2 Distributed Energy Resources . . . 12
- 2.3 Grid-interactive Buildings . . . 13
  - 2.3.1 DER Scheduling . . . 14
  - 2.3.2 Energy Trading . . . 14
- 2.4 GEBs And Load Forecasting . . . 16
- 2.5 Load Forecasting . . . 17
  - 2.5.1 Classical Forecasting . . . 18
  - 2.5.2 AI Forecasting . . . 19
- 2.6 Residential Load Forecasting . . . 21
- 2.7 Long Short-term Memory Networks . . . 22
- 2.8 Research Gap . . . 26

2.9 Chapter Summary 26

3 Methodology **28**
3.1 Declaration of Terms 28
3.2 Dataset Preparation 28
3.2.1 Dataset Selection 29
3.2.2 Data Cleaning 33
3.2.3 Normalisation 33
3.2.4 Feature Selection And Encoding 34
3.2.5 Training and Validation Data Selection 36
3.2.6 Test Data Selection 37
3.3 Neural Network Design 38
3.3.1 System Specifications 39
3.3.2 LSTM Architecture 39
3.4 Experimental Setup 42
3.4.1 Experiment A: Training Data Amount 42
3.4.2 Experiment B: Prediction Horizon and Training Data Duration 44
3.4.3 Experiment C: Training Data Resolution 45
3.4.4 Training and Validation Data Incrementation 47
3.5 Evaluation Metrics 48
3.5.1 Prediction Model Error 48
3.5.2 Goodness of Fit 49
3.5.3 Energy Prediction Error 49
3.5.4 Peak Power Prediction 50
3.5.5 Training Duration 51
3.6 Chapter Summary 51

4 Results And Discussion **52**
4.1 Experiment A: Training Data Size 52
4.1.1 Regression Analysis 53
4.1.2 Energy Analysis 55
4.1.3 Power Peak Analysis 57
4.1.4 Prediction Error Variance 58
4.1.5 Discussion 60
4.1.6 Summary of Findings 61
4.2 Experiment B: Prediction Horizon and Training Data Duration . . . 62
4.2.1 Model Prediction Error Analysis 63
4.2.2 Energy Analysis 64
4.2.3 Power Peak Analysis 65
4.2.4 Goodness of Fit 66
4.2.5 Model Training Duration 68
4.2.6 Discussion 69
4.2.7 Summary of Findings 71
4.3 Experiment C: Training Data Resolution and Training Data Size . . 72

4.3.1 Model Prediction Error Analysis 74
4.3.2 Energy Analysis 77
4.3.3 Power Peak Analysis 78
4.3.4 Goodness of Fit 79
4.3.5 Discussion 82
4.3.6 Summary of Findings 83
4.4 Chapter Summary 83

5 Conclusion And Future Work **85**
5.1 Introduction 85
5.2 Findings 86
5.3 Future Work 87
5.4 Conclusion 88

List of Figures

2.1 Outline of grid-interactive building application [42]. 13
2.2 Architecture of a single LSTM cell adapted from [93] 25

3.1 Box-plot analysis of seasonal power usage across 3 years. 32
3.2 Encoding of timestamp labels over a 24-hour period into a numerical signal using sine and cosine transforms. 35
3.3 Partitioning of dataset measurements into three contiguous training data subsets. 37
3.4 Test data allocations for the training-validation subsets of each year 38
3.5 Outline of the stacked-LSTM neural network architecture 40
3.6 Flow diagram of Experiment A algorithm 43
3.7 Flow diagram of Experiment B algorithm 44
3.8 Comparison of the same set of load profile measurements at 1 and 30-minute resolutions 46
3.9 Flow diagram of the Experiment C algorithm 46
3.10 Selection of measurements for the 1, 6 and 12-month sizes of training-validation data in the Year 1 subset. 48

4.1 Comparison of model prediction errors and training time against training data size 53
4.2 Comparison of predicted vs actual power and cumulative residual energy for 1, 90, 450 and 990-day prediction models. 56
4.3 Comparison of observed power peaks vs. predicted power profiles for the 1, 90, 450 and 990-day models. 57
4.4 Box plot analysis of the load profile datasets. Measurements begin in 2010 and extend backwards ending in 2007. 59
4.5 Comparison of model prediction errors and training data size for various prediction horizon lengths 62
4.6 Comparison of mean MAEs for models of each training data size. . 63
4.7 Comparison of predicted and observed power and energy for 1, 6 and 12-month training data sizes at a 1-hour prediction horizon . . 64
4.8 Comparison of observed peaks vs predicted power profile for the 1, 6 and 12-month prediction models. 66
4.9 Comparison of R^2 scores for 1, 3, 6, 9 and 12-month training size models over three years. 67
4.10 Comparison of training durations for each training data size up to a prediction horizon length of three hours 68
4.11 Comparison of mean model training time for each year of training data up to a horizon length of three hours. 68

4.12 Regression analysis of mean model training time for each year of training data. 69
4.13 Illustration of the MAE and R^2 scores within the 3 hour prediction horizon length. 70
4.14 Comparison of observed and predicted power between 2-hour and 10-hour horizon length models. 71
4.15 Comparison of model prediction errors and training data size for various data resolutions. 72
4.16 Mean Difference In MAE For Lower Resolution Models Relative to 1-Minute Resolution Models. 75
4.17 Box plot analysis of MAE distributions for models of each resolution. 76
4.18 Comparison of predicted and observed power and energy for 1, 5 and 30-minute resolution models. 78
4.19 Comparison of the observed power peaks and predicted power for the 1, 5 and 30-minute resolution models 79
4.20 Comparison of R^2 scores for 1, 5, 15, and 30-minute resolution models over 3 years. 80
4.21 Mean R^2 scores of each year for 1, 5, 15, and 30-minute resolution models. 81

List of Tables

3.1 Summary of load profile dataset characteristics. 29
3.2 Seasonal minimum, median and maximum power usage for each year of the training dataset. 32

4.1 Summary of select Experiment A results. 54
4.2 Summary of power and energy predictions for select models. 56
4.3 Summary of MPE values for select models. 58
4.4 Summary of MPE values for 1, 6 and 12-month models for 1-hour horizon. 65
4.5 Summary of power and energy predictions for select models. 78
4.6 Summary of power peak predictions for select models. 79

Abbreviations

AC	**A**ir **C**onditioning
AI	**A**rtifical **I**ntelligence
ANN	**A**rtificial **N**eural **N**etwork
AR	**A**uto **R**egressive
ARIMA	**A**uto **R**egressive **I**ntegrated **M**oving-**A**verage
CNN	**C**onvolutional **N**eural **N**etwork
CPP	**C**ritical **P**eak **P**ricing
DER	**D**istributed **E**nergy **R**esources
DSM	**D**emand-**s**ide **M**anagement
GEB	**G**rid-interactive **E**fficient-**B**uilding
GRU	**G**ated **R**ecurrent-**U**nit
IOT	**I**internet **o**f **T**hings
LSTM	**L**ong **S**hort-**t**erm **M**emory
MA	**M**oving **A**verage
MAE	**M**ean **A**bsolute **E**rror
MAPE	**M**ean **A**bsolute **P**ercentage **E**rror
ML	**M**achine-**l**earning
MLR	**M**ultiple **L**inear **R**egression
MPC	**M**odel **P**redictive **C**ontrol
PCDL	**P**hysically **C**onsistent **D**eep-**l**earning
PDI	**P**er-capita **D**isposable **I**ncome
PPR	**P**eak **P**ower **R**atio
PTP	**P**eer-**t**o-**P**eer
PV	**P**hoto-**v**oltaic
REC	**R**esidential **E**nergy **C**onsumption

RNN	**R**ecurrent **N**eural **N**etwork
RMSE	**R**oot **M**ean **S**quare **E**rror
RTP	**R**eal-**t**ime **P**ricing
SI	**S**prawl **I**ndex
SVM	**S**upport **V**ector **M**achine
TOU	**T**ime **o**f **U**se
UHI	**U**rban **H**eat **I**sland

Publications

Peer-reviewed conference and journal papers directly related to this Dissertation.

Simani, K. N. Genga, Y. O. and Yen, Y. C. J (2023), "Using LSTM To Perform Load Modelling For Residential Demand Side Management", *Proceedings of the 31st Southern African Universities Power Engineering Conference*, http://dx.doi.org/10.1109/SAUPEC57889.2023.10057875

Simani, K. N. Genga, Y. O. and Yen, Y. C. J (2024), "Using LSTM To Perform Load Predictions For Grid-Interactive Buildings", *SAIEE Africa Research Journal*, http://dx.doi.org/10.23919/SAIEE.2024.10520212

CHAPTER 1

Introduction

The global demand for electrical energy is on the rise and is expected to continue increasing, with an average increase of 3.2% predicted for the period of 2024-2025 [1]. Approximately 27% of the global energy demand is attributed to consumption in the residential sector: water heating, thermal comfort, cooking and lighting [2]. Several factors contribute to the sustained growth of residential energy demand including increasing population size, climate change and urbanisation.

The residential sector represents millions of flexible loads that can be more effectively managed to improve efficiency and reduce energy consumption. Governments and utilities employ various demand-side management (DSM) strategies such as time-of-use (TOU) tariffs to actively manage consumption by influencing consumer usage behaviour [3]. Yet, the efficacy of DSM is limited by its reliance on voluntary consumer participation.

Grid-interactive efficient buildings enable automated participation in energy management strategies by dynamically shifting household load in response to grid fluctuations and DSM signals. Furthermore, GEBs can manage interactions between distributed energy resources (DER) and the grid to minimise consumer costs and enable the buying and selling of credits through energy trading. The addition of accurate load demand forecasting would enable GEBs to perform fully autonomous decision making to manage energy, optimise costs and energy trading and ensure user comfort.

However, accurately predicting residential load demand is a challenging task due to its highly stochastic nature. Machine-learning algorithms are a tool commonly used to perform predictions on time-series data such as residential loads and are

well suited to forecasting load profile patterns. These algorithms can be used to produce predictions that are multiple time-steps into the future.
The development of a performant predictive machine-learning model traditionally involves an iterative, trial-and-error process that requires the selection and tuning of various parameter values to train and evaluate numerous prediction models. This can be a time-, data- and compute-intensive procedure. In the practical setting of a GEB home the energy and processing hardware required to complete an extensive model development cycle could incur large additional costs to the consumer. In addition the tuning of certain parameters have implications such as requiring the purchase of data capture devices.

1.1 Problem Statement

It is well established that using more data i.e., data resolution and size, improves model prediction accuracy and the ability to forecast further ahead [4–7]. In practice however, there are costs and limitations of model development and parameter tuning including: real-time collection and storage of load profile measurements, cost of measurement devices and finite compute power. Therefore it is important to identify the extent to which additional data is required to achieve performant model fidelity while balancing the time and resource requirements of real-world model development. This work investigates predictive model resource trade-offs to inform model design decisions.

1.2 Approach

This dissertation investigates to what extent the machine-learning model design process can be simplified by determining generalisable relationships between specific parameter values and their impact on model performance to inform design choices. Literature exists on a variety of optimisation techniques for model tuning as well as hyperparameter and data feature selection [8–10]. Yet, there is a lack of research regarding the practical implications of implementing a real-world

machine-learning load forecasting model.

For this work a long short-term memory (LSTM) machine-learning algorithm is used to produce a series of load forecasting models in three experiments each investigating a specific model design parameter including: training data size, prediction horizon length and data resolution. For each experiment LSTM prediction models are trained using a range of values for the particular parameter. The performance of each model is evaluated using a baseline 24-hour period of load profile data during the summer solstice for three consecutive years. To determine the relationships between the design parameters and model performance, the model output is compared against the parameter values used to develop it.

1.3 Research Question and Objectives

Therefore this work aims to address the question: *To what extent can the relationship between machine-learning design parameters and their predictive model fidelity inform model design choices for practical load profile applications?* The following three objectives are derived from this question:

1. Investigate the relationships between training data size, prediction model error and compute intensity.
2. Investigate the relationships between prediction horizon length, training data size, model prediction error and compute intensity.
3. Investigate the relationships between data resolution, training data size, and prediction model error.

This dissertation is structured as follows:

In *Chapter 2* the background and literature review are presented. The global demand for energy and its relationship to growing populations is discussed. Various DSM strategies to address the energy demand are reviewed. The potential for GEB homes to improve residential energy efficiency and augment DSM efficacy is

explored. Lastly, the role of machine-learning and load forecasting in extending and optimising GEBs functionality is discussed.

Chapter 3 outlines the machine-learning model design and experimental approach for this work.

In *Chapter 4* the results from each of the three experiments are analysed and discussed.

Chapter 5 summarises the findings and details future work. An evaluation of the experimental results in relation to the stated research objectives will be used to conclude on this dissertation.

CHAPTER 2

Background and Literature Review

This chapter provides background information on the current state and outlook of residential energy demand and management in section 2.1. Various DSM strategies aimed at reducing residential demand are then detailed in section 2.2. In section 2.3 the role of GEBs in residential energy management and DSM automation is discussed. Then, section 2.4 examines the role of load forecasting in augmenting GEB functionality. Furthermore, traditional and machine-learning based load forecasting methods are introduced and compared in section 2.5. Lastly in section 2.8, the lack of research exploring the requirements of developing machine-learning models for practical load forecasting applications is discussed.

2.1 Residential Energy Demand

The global demand for electrical energy is on the rise and is expected to continue increasing. During the period of 2015-2019 the average increase of global electricity demand was 2.4% [1]. In 2023 this demand rose to 2.6% and is predicted to rise to an average of 3.2% in the period of 2024-2025. Approximately 27% of global energy demand is attributed to consumption in the residential sector [2]. This includes energy used for space heating and cooling, residential appliances, water heating, cooking and lighting. There are several factors that contribute to the sustained growth in demand including urbanisation, economic and population growth as well as the lack of electrification in developing countries.

2.1.1 Urbanisation

Urbanisation is a driver of increased electrical energy consumption. Evidence from literature shows that there is a strong correlation between high rates of urbanisation and the increase in electrical energy consumption in residential buildings.

In an analysis of provincial population data in China, it was shown that for a 1% increase in the population of medium to large cities there was a 1.3% increase in residential energy consumption [11]. This population increase was attributed specifically to the migration from rural to urban centres instead of the natural growth of the present urban population. The degree of energy consumption by residents was shown to increase with the rate of urbanisation. For an urbanization rate less than 55%, a 1% increase in the urbanization rate resulted in a 0.09% increase in the energy consumption of residents. For an urbanization rate greater than 55%, a 1% increase in urbanization led to a 0.16% increase.

In India, there was a 73% increase in the urban population from 217 to 377 million in the period from 1991 to 2011 [12]. In this same period the national energy demand more than doubled from 320 million tonnes of oil (Mtoe) to 669 Mtoe despite a low per-capita energy consumption. The residential energy demand for India is projected to reach 230 Mtoe by 2030.

A consequence of high urbanisation rates is the necessity for increased usage of space for residential/housing purposes. This enables rapid geographic expansion of cities for the development of low-density, homogenous, residential housing otherwise known as urban sprawl [13]. The degree of urban sprawl has been linked to increases in residential energy usage. A sprawl index (SI) is defined by Zhao et al that relates built-up area growth as a ratio of the population growth [14]. The authors show that for a higher SI there is an increase in energy use per-capita. This was demonstrated by comparing the SI of two cities in China for the period of 2001 to 2010. During this period in the city of Chongqing the SI increased to 2.74 and the per-capita energy consumption increased by 192%. For the same period in Huangzhou the SI increased to 0.61 but experienced an energy consumption growth rate of only 1%.

The large-scale migration from rural to urban centres is largely motivated by a desire for greater economic opportunities as people move from poorer rural areas to relatively wealthier urban areas with access to better jobs, higher wages etc [15–17]. Rapid urbanisation is linked to economic growth and an increase in disposable income for the urban population [18, 19]. An increase in average disposable income is associated with an increase in residential energy consumption. In a study across 33 provinces in Indonesia from 2010 to 2018, it was estimated that for every 1% increase in population density there is a 0.77% increase in electricity consumption [20]. This rise in electrical consumption was attributed to the increased living standards of the growing Indonesian middle-class and with it an increase in consumption of appliances for lighting, space cooling, and entertainment.

In [21] Wang et al demonstrate a link between the per-capita disposable income (PDI) of residents in urbanised China and the change in their residential energy consumption (REC). For lower levels of PDI residents show a decrease in their REC. However for residents with PDI above a certain average their REC instead increases. The increase in REC with PDI is attributed to the shift in economic status. As the PDI in the Chinese population increases, there is a shift in consumption habits from low-income, to middle and high income populations.

2.2 Residential Energy Management

To account for a growing residential energy demand there exist various strategies to manage grid usage, shift residential load demand and improve household energy efficiency.

2.2.1 Demand-side Management

DSM describes several activities and strategies undertaken by electrical utilities, governments and consumers to efficiently utilise electrical energy by actively managing consumption and the end user behaviour. The general aim of DSM is to

improve grid stability and performance and reduce energy consumption by monitoring and shifting load, either autonomously or by influencing changed behaviour in the consumer.

DSM can be broadly categorised into 4 main types: energy efficiency, demand response, dynamic demand and DER management [3]. For DSM, energy efficiency means educating consumers to make more informed decisions so that they are more likely to purchase or incorporate more energy-efficient high load appliances. Demand response involves methods to change the the load profile shape such as: reducing the grid load during the peak demand period (peak shaving), increasing grid load during off-peak periods (valley filling) and shifting grid load from the peak to off-peak periods (load shifting). Dynamic demand exploits appliance operating cycles to improve grid power factor. Lastly, DER management includes management of generation and storage of distributed and renewable energy sources.

For the utility this improves grid stability by reducing the risk of grid failure from excessive peak demand, and defers or reduces additional grid expansion investment by slowing the growth of peak demand [22]. The consumer benefits from reduced energy costs.

DSM is achieved by monitoring load profile patterns and influencing consumers to change their consumption behaviour which can be supported by device timers or autonomous control switches. The implementation of dynamic energy pricing schedules through TOU tariffs are one of the primary means for utilities to incentivise consumers to modify their consumption behaviour [23]. Additionally, as the proliferation of DER in residential spaces is becoming more common, DSM has expanded to include the balancing and integration of variable generation sources such as rooftop PV with the grid [24].

Dynamic Energy Pricing

As a DSM tool, dynamic energy pricing emphasises the use of tariffs as financial incentive for consumers to reduce their energy consumption or shift their energy usage to different periods of the day. These tariffs can be grouped into one of

two categories: dynamic tariffs and load curtailment. Dynamic tariff schemes are designed so that electricity is more expensive during periods when higher demand is expected and cheaper during periods of reduced demand. There are various types of dynamic tariffs including: time-of-use-pricing (TOU), critical peak pricing (CPP) and real-time pricing (RTP). The aim of this pricing is to incentivise consumers to modify their energy consumption patterns depending on the time of day thereby dynamically shaping the load profile.

Under TOU, tariffs are defined for set periods of the day for different seasons within a specified tariff cycle (e.g. start of the calendar year) and remain static during that cycle. TOU tariffs can be opt-out, whereby customers are automatically enrolled to the tariff and must actively request to be removed, or opt-in where customers voluntarily select and participate in the tariff. In general, these tariffs are structured such that electricity is more expensive during peak demand periods and cheaper in off-peak periods.

Due to their static nature, TOU tariffs are less reflective of the variations in cost of the electrical supply as compared to more dynamic tariffs such as RTP. However, consumers often prefer and are more responsive to simpler tariff structures [25]. In a meta-analysis of 27 TOU uptake studies Nicolson et al determined that the use of real-time pricing tariffs correlates with a 13% reduction in tariff uptake compared to static TOU tariffs. TOU tariffs are simpler than RTP tariffs and offer less flexible pricing but are also easier for consumers to understand. Yet as a consequence users are more likely to engage with the simpler TOU tariffs[26].

Grunewald et al demonstrate the trade-off between pricing signal complexity and consumer messaging [27]. They illustrate a scenario in which the ease of consumer messaging from static TOU tariffs could result in 4% to 8% savings in energy. In the context of high-volume wind farms Grunewald et al show that customers more readily engage with simpler, easier to understand tariffs. In a strong-wind environment where rapid demand responses to generation are necessary, having consumers more consistently engage and respond to changes in generation has a beneficial effect despite less accurate pricing information. Even among a selection of comparable TOU tariffs, consumers will tend to select the simplest option.

Where TOU tariffs have been widely implemented their ability to effect reductions on load demand and provide savings for customers has shown mixed results. Torrinti et al conducted an analysis of consumption behaviour in Northern Italy using data from 1446 residential meters over two years [28]. It was found that in the time following the introduction of an opt-out TOU tariff, average consumption increased by 13.69% while electricity bills decreased by 2.21%. Additionally, 75.6% of substations experienced an increase in electricity demand during peak periods. According to Torrinti, energy consumption is more susceptible to societal and daily demands and TOU tariffs alone do not deter from that pattern of usage. A study of 6957 electricity bills from households in Victoria, Australia showed that a 1% increase in peak-period prices relative to prices in the off-peak period resulted in a 0.2% shift in consumption from peak hours to off-peak hours [29]. Furthermore it was found that homes of the lower socio-economic consumers showed no responsiveness to TOU tariffs. Similar findings from by Yunusov et. al. demonstrates the distributional effects of TOU tariffs [30] . The impact of various tariffs imposed on smart meter readings from two UK smart meter data sets from 2011-2014 were analysed. Yunusov shows that the benefits of a given TOU tariff experienced by customers can vary based on their region, household composition and socio-economic status. They show that for a given tariff, low-income users in North East England would be the most financially advantaged while in London the same tariff would be most beneficial to high-income customers. According to Yunusov static TOU tariffs are unable to reflect variations in regional peak demands as well as lifestyle, occupancy and working patterns specific to different groups.

A shortcoming of TOU tariffs is that they do not provide additional incentives to reduce demand further during periods of elevated peak demand [31]. This is because they only reflect long-term average expectations of daily peak marginal costs and cannot account for seasonal changes or situational spikes in load demand [32]. CPP tariffs are designed to augment the static TOU tariff by including dispatchable surcharges to existing tariffs during limited periods of severe load demand. CPP tariffs are time-varying, opt-in tariffs whereby utilities charge higher prices

during specific, peak periods throughout the year. In exchange customers receive larger discounts on their electricity usage throughout the rest of the year outside of these critical periods. The aim of CPP is to more closely align the pay rate with the cost of producing electricity on the select few, most expensive days of the year.

CPP tariffs have been shown to achieve significant load reduction during peak periods. In an analysis of residential customer response to CPP tariffs in California Herter et. al. found that during 5-hour critical peak periods CPP participants used up to 13% less energy on average than they did during normal peak periods [33]. The CPP prices were about three times the price of the applicable peak-time TOU tariff. Kato et al conducted an experimental CPP trial using homes in the City of Kitakyushu, Japan [34]. They found that he number of households practising maximum-saving behaviour as a result of the CPP tariffs increased by between 6.5% and 8.8%. Of the total electrical energy saved during the CPP period of this trial, 50.4–60.9% was attributed to maximum-saving behaviours of consumers. In an analysis of 15 different price response experiments conducted in the United States, France, Australia and Canada, Faruqui et. al. determined that TOU rates induce a drop in peak demand that ranges between 3 and 6% while CPP tariffs induce a drop in peak demand that ranges between 13 and 20% [35].

RTP tariffs feature more granular pricing schemes that vary on hourly or even sub-hourly periods. The price levels are updated by the utility and the changes are communicated to customers a short period in advance. Compared to static TOU tariffs, RTPs most closely reflect the continuous, real-time cost of electricity. RTPs are designed to change with the real-time variations in electricity production costs through continuous pricing level and tariff period updates. This enables consumers to take advantage of price fluctuations throughout the year. For utilities RTP allows for dynamic responses to variations in the demand-generation balance caused by incidental as well as seasonal changes. Consumer consumption behaviour can be shifted and adjusted according to the current generation environment.

2.2.2 Distributed Energy Resources

The uptake of distributed energy resources (DER) in residential spaces has risen in recent years [36]. These include renewable energy sources such as rooftop photovoltaic (PV) arrays, residential wind farms, natural gas generators etc. By using DERs consumers are able to independently produce energy with fewer carbon emissions and reduce the amount of electricity purchased from the utility/grid.
In the context of DSM the use of DERs enables consumers to more effectively participate in TOU, especially load curtailment tariffs. These tariffs compensate customers for reducing their load during peak loading periods. In a case study of a residential energy hub model for a home incorporating a TOU tariff, Rastegar et al. show that the inclusion of PV leads to 61% and 65% reduction in daily and annual tariff costs respectively. They found that if solar panels exist at a home, operation of appliances shifts to the cheaper tariff hours when it is sunny and the PV output is at its highest. In addition, a study on the impact of DERs on selected utility distribution feeders in Colorado, U.S Paudyal et al. found that electricity costs for residential consumers with PV and battery storage systems were reduced on average by 20.7% when those DERs were combined with TOU rates [37].
A barrier to the adoption of residential DER systems is the high investment cost e.g., solar panels, battery storage and inverters for a PV system [38]. In order to offset the investment cost and maximize potential cost savings it is necessary to implement efficient DER coordination and scheduling techniques to take advantage of TOU and RTP tariffs. Pedrasa et al. implement a particle swarm optimization tool that determines the optimal choice between simultaneous or individual DER scheduling by comparing the end-user cost of each approach [39]. They show that in some cases a simultaneous DER approach can produce increased net benefits, while for others coordination does not add significant value. Omu et al. create a mixed integer linear programming (MILP) model to design a DER system to meet electrify and heating needs for a collection of residential and commercial buildings while minimising investment and operation costs [40]. In a case study of buildings

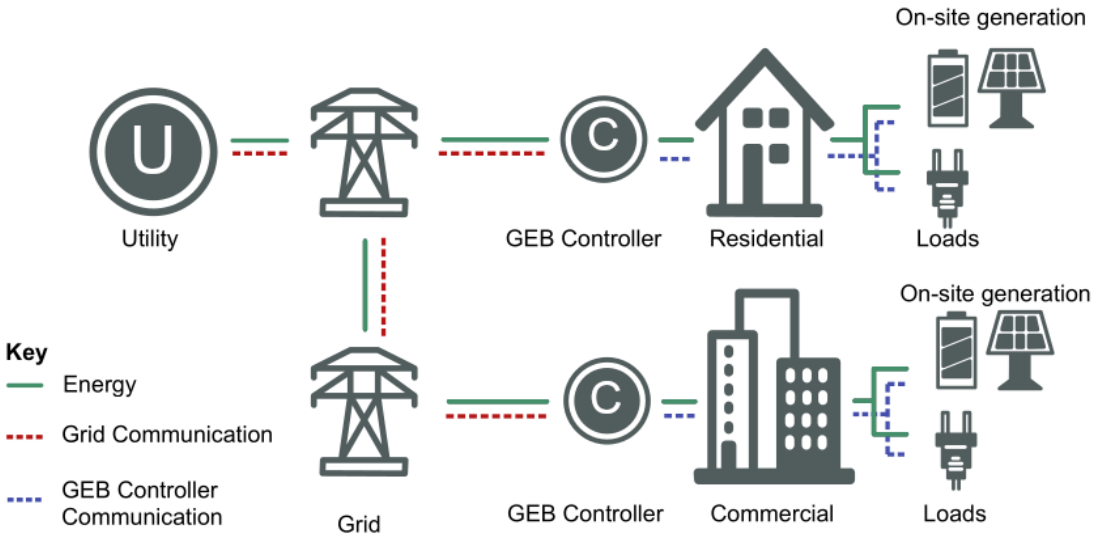


FIGURE 2.1: Outline of grid-interactive building application [42].

located in South England their model produced an optimised DER system that was found to decrease annual costs by 40% and carbon emissions by 50%.

2.3 Grid-interactive Buildings

Grid-interactive efficient buildings (GEB) incorporate smart devices and internet-of-things (IOT) technologies to connect, monitor and control devices within a household while also communicating their energy usage information to electrical utilities through smart metering [41]. Figure 2.1 illustrates an outline of a GEB network. Within a GEB household possible future developments could see the automated management of energy as pricing signals and energy consumption will be communicated directly between the utility and appliances without requiring additional consumer input [41]. Device operation can be automatically coordinated to shape the residential load profile in response to changing energy consumption, generation and pricing requirements [43]. This would enable utilities to more effectively implement DSM strategies and allow consumers to more conveniently and efficiently participate in DSM. Fully automated GEBs are envisioned to participate in optimised, dynamically shifting load and responding to grid fluctuations for utility DSM strategies [44]. GEBs with on-site power generation and storage,

could manage interactions with the grid for buying or selling of credits through energy markets.

2.3.1 DER Scheduling

In a GEB the scheduling of DER and grid energy usage can be intelligently coordinated to take advantage of dynamic energy pricing. By using DERs during periods of peak energy costs consumers can minimise grid usage and maximise savings while still having full access to their devices. Then, during off-peak periods grid energy can be sourced while the DERs replenish. Traditionally this scheduling was manually implemented by the consumer but GEB automation can enable this process to occur automatically and dynamically based on available DER generation, energy requirements and TOU pricing signals.
Liu et al. developed an action dependent heuristic dynamic programming (ADHPD) algorithm to optimise the scheduling of a residential PV-battery system under an RTP tariff that adapts the scheduling strategy based on current weather conditions [45]. Simulations of their ADHDP system showed 30.7% in savings from reduced grid energy costs. Pooranian et al. proposed a MILP optimization model for jointly minimizing electricity costs and carbon emissions by considering consumer preferences in a smart residential building equipped with a PV-battery system, composed of 30 smart homes and under two different price tariffs [46]. Their simulation results demonstrated that the proposed energy management model resulted in lower energy and carbon emission costs and was able to reduce the peak demand for the building.

2.3.2 Energy Trading

Traditionally electrical energy was consumed unidirectionally: electricity is produced at the generation side, dispersed through the distribution network and then ultimately consumed by the end-user. However, with the penetration and improvements in residential DER technology, especially PV, residential consumers

are now able to generate energy in excess of their daily requirements. This surplus can be utilised to transact with other consumers or sold back to the utility through the grid, creating potential for residential energy trading [47].

Yet performing effective energy trading is time and computation intensive, requiring constant real-time monitoring of the energy consumption for all household devices as well as the available energy and rate of DER generation [47]. Furthermore, participating in the energy market would require facilitating numerous transactions between a multitude of other consumers and DER environments.

The GEB is positioned as an ideal environment for real-time energy trading. The IOT infrastructure of GEB include the device communication, connectivity and data processing that would enable continuous DER generation and home consumption monitoring. GEB can autonomously perform buy/sell transactions on the energy market by directly interfacing with the GEB energy management systems of other participants.

Nizami et al. proposed a neighbourhood energy management system that mitigates the demand peaks and provides electricity bill reduction to the participating consumers [48]. This optimization model minimizes the energy cost reducing aggregated demand peaks in the neighbourhood consumption by scheduling the charge-discharge operation of energy storage units and energy sharing among participants. Using a case study for a residential neighbourhood in Sydney, Australia their simulation results show that the system can effectively reduce neighbourhood peak demand peaks and participants can receive profits of up to 4$ per day by trading excess storage energy with neighbours. The feasibility of applying peer-to-peer (PTP) energy trading to reduce costs for energy consumers and to increase income for DER producers in a community micro grid was investigated by Long et al. [49]. They examined the business models of three different PTP market paradigms and assessed their ability to reduce consumer energy costs and increase income. It was found that with a moderate level of PV penetration, PTP energy trading resulted in a reduction of community energy costs by approximately 30%.

2.4 GEBs And Load Forecasting

The described energy management activities that can be performed within a GEB are each dependant on some temporal component. For energy trading, like any market, it is critical to predict future energy supply and demand in order to optimise transaction returns and minimise surplus household energy [50]. Effective DER scheduling requires coordinating the limited periods of energy generation with consumer requirements and the dynamic pricing of TOU and RTP tariffs. The ability to forecast load would enable more efficient timing and allocation of DER to meet satisfy consumer needs while also reducing energy costs [51]. Therefore, to fully exploit the dynamic energy management capabilities of GEB homes the ability to forecast load demand is crucial.

A study by Hassan et al. examines the abilities of three different models to forecast a building's energy consumption for a PTP energy trading network [52]. They showed that forecasting energy demand with accurate results is achievable and has significant benefits for PTP energy trading, including lower system costs, reduced losses, and improved reliability of the network. Similarly, Manchalwar et al. implemented and compared objective PTP energy trading networks using various integrated short-term load forecasting methods. They found that the total energy cost in their PTP trading network was reduced by 36.14% with the inclusion of short-term load forecasting.

Iwafune et al. implemented a PV-battery energy management system using day-ahead electricity consumption forecasting methods [53]. Ten forecasting methods were examined using real survey data from 35 households over the period of one year. Using a multiple linear regression method to forecast consumption and maximise PV energy usage, they found that the annual energy cost of a simulated home could be reduced by up to 10%. A long-term power forecasting method is proposed by Zhao et al. which predicts future power usage to determine optimal energy storage sizing for a DER [54]. They use a hybrid load forecast model to predict the net power signal four years ahead based on ten years of historical power data. An optimization algorithm uses the predicted power is to determine

an optimal energy storage capacity. From test results on Western Australia power grid data it was shown that the authors' proposed forecasting method achieves a maximum error reduction of about 63.1% compared to traditional methods and reduced energy storage installation costs by 56.4%.

2.5 Load Forecasting

Electrical load forecasting is the estimation of the magnitude of power demand for a geographical location over specific periods of time. It has generally been used by electrical utilities and industries to anticipate future load patterns on the generation and distribution level. This would enable better informed investment decisions and planning for future electrical infrastructure as well as minimise the risk of supply shortages. Load forecasting can be divided into short-term (an hour to one week), mid-term (a week to one year) and long-term periods (one to ten years) [55–57]. Charytoniuk proposed a short term forecasting approach that was able predict the total load of a power utility up to 90 minutes ahead [58]. Ghiassi et al performed monthly, quarterly and yearly forecasts of the total power sales of the Taiwan Power Company [59]. Willis et al use regression models to determine medium-term extrapolations of peak feeder loads within a five-year period [60]. Cetinkaya used artificial intelligence and mathematical modelling to forecast energy consumption and peak loads in Turkey over a 20 year period [61].

Electrical load forecasting can be broadly categorised into two methods: parametric and artificial intelligence methods. Parametric methods use the relationships between load demand and its determining factors such as geographical location, population demographics and economic data to form mathematical models of the load profile [61–63]. The model parameters are then estimated by applying statistical techniques on electrical load, economic and demographic data. Artificial intelligence (AI) methods are a more recent load forecasting approach that refers to a broad set of techniques that leverage machine-learning and deep-learning algorithms to determine the load prediction models [61–63].

2.5.1 Classical Forecasting

Electrical load forecasting was traditionally performed using statistical methods and techniques [62–64]. Using linear mathematical processes, these methods require the forecaster to explicitly identify, define and model the relationships between the load and the variables that influence its change. Some of the frequently used techniques include: linear regression, multiple linear regression and autoregressive integrated moving-average (ARIMA) models [62–64].

The linear regression method models the functional relationship between the load demand weather and a single independent influencing factor such as weather conditions, time of day and customer demographic [62, 65]. This method assumes that the predicted load can be decomposed into a standard load component and a component that is linearly dependent on the influencing factor. The model is expressed as:

$$L(T) = L_n(t) + \sum a_i x_i(T) + e(t) \tag{2.1}$$

Where $Ln(t)$ is the standard load at time t, a_i is the estimated varying coefficients, $x_i(T)$ are the independent influencing factors (e.g. weather), $e(t)$ is a white noise component and n is the number of observations.

Multiple linear regression (MLR) is an extension of the linear regression model that uses ordinary least squares estimation to determine a linear relationship between a single dependant variable (i.e. the load) and multiple independent variables (i.e. load influencing factors). The model is represented as follows:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_n X_n + e \tag{2.2}$$

Where Y is the load value, $X_1, X_2, X_3 ... X_n$ correspond to n independent explanatory variables, $\beta_0, \beta_1, \beta_2, \beta_3 ... \beta_n$ are the regression coefficients and e the error between predicted and observed load values.

ARIMA models are a statistical analysis tool that are among the most popular approaches to forecasting time-series load data [66]. Time-series data are a series of data points measured at consistent time intervals where each data point

in the series is dependent on the previous data points i.e. past data points are explanatory variables for future data points. An ARIMA model is characterized by 3 terms p, q and d, where p is the order of the AR (auto-regressive) part of the model term, q is the order of the MA (moving average) term and d is the order of differencing required to make the time series stationary. The general form of the ARIMA model can be expressed as:

$$\phi_p(B)(1-B)^y Y_t = \theta_q(B)e_t \tag{2.3}$$

Where $\phi_p(B)$ is the moving-average operator, $\theta_q(B)$ is the autoregressive operator, Y_t is an observation at time t and e_t is the prediction error distributed as white noise.

Juberias et al use ARIMA to develop an hourly load forecasting model as part of a real-time control system for Spain's national grid operator [67]. They were able to achieve an average error of less than 2% with forecasts up to 5 hours ahead.

Although statistical methods have seen extensive use in load forecasting, their implementation can be time and resource intensive as they require specialised, expert knowledge to manually develop effective models for each problem domain. Furthermore, because these statistical methods use linear processes they perform poorly when modelling datasets that include non-linear relationships and dependencies [64, 68].

2.5.2 AI Forecasting

More recently, AI techniques have seen extensive applications in load forecasting research [7, 69–71]. AI load forecasting methods include a variety of machine-learning and deep-learning algorithms that are able to independently determine dependencies and patterns between variables within a dataset. By using historical data of the problem domain as input, AI forecasting can determine models that relate variations in explanatory variables such as weather, time and temperature to future outcomes of the load. Compared to statistical methods AI models are

relatively more accessible as their algorithms are purely data-driven and do not require expert knowledge of the system to implement an effective model e.g., determining system models, order differencing, data stationarity etc. Additionally AI models have been shown to be well-suited to forecasting data with non-linear dependencies [64, 68, 69]. Among the most commonly used AI forecasting models are the support vector machine (SVM), and the artificial neural networks (ANN) [7, 69, 71].

The SVM is a supervised machine-learning algorithm that solves regression problems by performing optimal data transformations to determine boundaries between data points based on predefined classes [72]. The main objective of the SVM algorithm is to find the optimal hyperplane in an n-dimensional space that distinguishably segregates the data points of different classes in the feature space. In a comparative study by Mitchell et al, ANN and SVM techniques were applied to perform short term (one hour) load forecasts of a small island power system in Trinidad and Tobago for three load types:batch, continuous and batch-continuous [73]. Each load type was representative of a unique industrial customer. They showed that, for a continuously varying load, the SVM yielded a performant model with a maximum mean absolute percentage error (MAPE) of 1.20%. Fu et al applied an SVM to perform next-day load predictions of a public building using disaggregated load data. Their SVM model demonstrated the least prediction errors, outperforming 3 other benchmark regression models including an ARIMAX, decision tree, and ANN [74].

An ANN is a mathematical model that attempts to emulate the structure and functionality of biological neural networks [75]. The fundamental building block of the ANN is an artificial neuron that is represented by a mathematical function. By connecting multiple layers of connected neurons a neural network is formed. The ANN can be seen as a series of non-linear statistical models that can model complex relationships between inputs and outputs by learning variable dependencies and patterns from a sample set of data. ANNs encompass multiple variants including but not limited to, convolutional neural networks (CNN) and recurrent neural networks (RNN) [76, 77].

To address the complexity of forecasting variable power sources in a micro grid setting Hernandez et al proposed an ANN-based model for short-term load forecasting in a disaggregated, micro grid-sized environment [78]. By carefully selecting the patterns within the test data their ANN prediction model was able to achieve mean prediction errors as low as 3%. A day-ahead, CNN forecast model for the half-hourly load of Victory, Australia was proposed by Khan et al [79]. To forecast the total load for one day of the week, they trained the CNN model on load data for that same day over the previous three months. The CNN model's performance was compared against 4 other model types and found to produce the lowest MAPE, mean absolute error (MAE) and root mean square error (RMSE) values of 2.1%, 138.771 and 116.417 respectively. An RNN model to produce daily residential load forecasts was proposed by Shabbir et al [80]. Their RNN model demonstrated a 15% reduction in RMSE when compared to an SVM machine performing a similar prediction.

2.6 Residential Load Forecasting

From the discussion in section 2.3, as residential smart metering and inter-appliance communication through IOT becomes more commonplace, large volumes of high-resolution, time-series, domestic power consumption and generation data is becoming increasingly more accessible. As mentioned in section 2.5.2, AI load forecasting methods are designed to produce predictive models by training on large samples of load data to determine the complex patterns and dependencies between variables in the data. In addition, these models are more convenient to implement compared to the traditional statistical methods that require expert knowledge and time to develop models specific to each use case. These traits make AI load forecasting well suited to forecasting residential load profiles which can consist of large datasets containing numerous profile patterns specific to each household [81–83]. Furthermore, several AI load forecasting models have been shown to perform residential load forecasting better than the traditional techniques.

To address the high degree of non-linearity between residential input data and

output predictions, Sajjad et al implemented a hybrid CNN-gated recurrent units (GRU) model to perform short-term load forecasts of residential buildings [84]. Their model was tested on two publicly available datasets and compared against benchmark models. The CNN-GRU model attained the lowest values of 0.22, 0.47 and 0.33 for MSE, RMSE and MAE metrics respectively as compared to other models including linear regression and SVM. Wen et al proposed a hybrid RNN-GRU hourly forecast model to achieve high forecasting accuracy with limited input variables [85]. The RNN-GRU model was evaluated using aggregated and disaggregated residential building load demand data and compared against several other benchmark methods. Their model demonstrated the lowest prediction errors, outperforming conventional methods such as ARIMA, SVM and MLR. Furthermore the model was able to impute missing values within the training dataset. To overcome the occurrence of over-fitting due to additional neural network layers Shi et al proposed a novel pooling-based deep RNN that batches household load profiles into a pool of inputs [86]. Their method was evaluated using smart meter data from 920 households in Ireland, and compared against conventional household load forecasting techniques, outperforming ARIMA by 19.5%, SVR by 13.1% and classical deep RNN by 6.5% in terms of RMSE.

2.7 Long Short-term Memory Networks

RNN models are among the most widely used for residential load forecasting applications [87, 88]. Unlike other neural network architectures, RNNs are able to retain memory of their neuron calculations which enables them to identify long-term dependencies in sequential data. As a consequence of their long-term memory capabilities, standard RNN models experience the vanishing gradient problem [89]. During training, the RNN learns by adjusting the weights of its neuron connections to minimize the difference (i.e., error) between the predicted and observed outputs. This process is achieved through a technique called backpropagation, where the gradient of the loss function with respect to the neuron weights is calculated and used to update the model parameters[90]. Vanishing gradients occur

when the gradients computed during backpropagation become extremely small as they propagate backward through the neural network layers. Consequently, for longer data sequences the weights in the early layers of the network receive very small updates that either significantly slow down the rate of network learning or prevent it altogether.

The long short-term memory (LSTM) neural network is an improvement on the RNN that was specifically designed to avoid the vanishing gradient problem by incorporating gating functions to control the flow of information and changes in the network gradients [91]. There are 3 types of gates included in an LSTM: the input gate, the forget gate, and the output gate. The forget gate selectively removes information from the cell state, the input gate determines what new, relevant information from the data is stored in the cell, and the output gate decides the value that is output by the cell. These gates enable LSTMs to control the flow of gradients through the network, effectively addressing the vanishing gradient problem, and allows the network to learn and retain information over longer sequences.

The structure of an LSTM network consists of a series of cells, each of which consists of a set of gates that control the flow of information into and out of the cell. Each LSTM cell also contains a memory unit that stores information of the cell's state from previous time steps that is used to determine the output of the cell for the current time step. Figure 2.2 depicts the architecture of a single LSTM cell. As previously mentioned, the gates are used to selectively forget or retain information from previous time steps allowing the LSTM to determine the most relevant long-term dependencies in the input data. Each gate processes information by using a sigmoid activation function to output a value between 0 and 1: a value of 0 meaning all of the information is blocked by the gate and a value of 1 meaning all of the information is passed through the gate [92, 93]. The equations for each of the 3 gate types are as follows:

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.4}$$

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.5}$$

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.6}$$

Where f_t, i_t, o_t represent the forget, input and output gates respectively. σ represents the sigmoid function, w_x is the neuron weight for the respective gate x associated with each LSTM cell. h_t is the previous output at timestamp $t-1$, x_t denotes the current input vector at timestamp t and and b_x is the neuron bias for gate x.

The equations to determine the current cell state, the next cell state and the cell output are defined as follows:

$$\tilde{C}_t = tanh(w_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.7}$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{2.8}$$

$$h_t = o_t \times tanh(C_t) \tag{2.9}$$

Where C_t and C_{t-1} represent the new and previous cell states (memory) at timestamps t and $t-1$ respectively. $\tilde{C}_t$ is the output of the tanh activation function at timestamp t and $\times$ is the element-wise multiplication of the vectors. The output of each LSTM cell is passed to the next cell in the network, enabling the LSTM neural network to process and analyse sequential data over multiple long-term and short-term time steps [92–94].

The ability to retain long and short-term dependencies make the LSTM suitable for long, complex, time-series patterns such as electrical load profiles [95, 96]. The LSTM has been applied extensively in load forecasting as well as energy trading, DER management and GEB research.

Bouktif et al proposed a Genetic Algorithm (GA) optimized, LSTM model for short to medium-term aggregate load forecasting [8]. Using metropolitan electricity consumption data from France as a case study, their LSTM model was found to outperform 7 other machine-learning forecasting models. A novel Physically Consistent Deep Learning (PCDL)-based optimization framework for day-ahead energy dispatching and thermal comfort control within a grid-interactive community was proposed by Xiao et al [97]. Using an LSTM forecast model, the PCDL

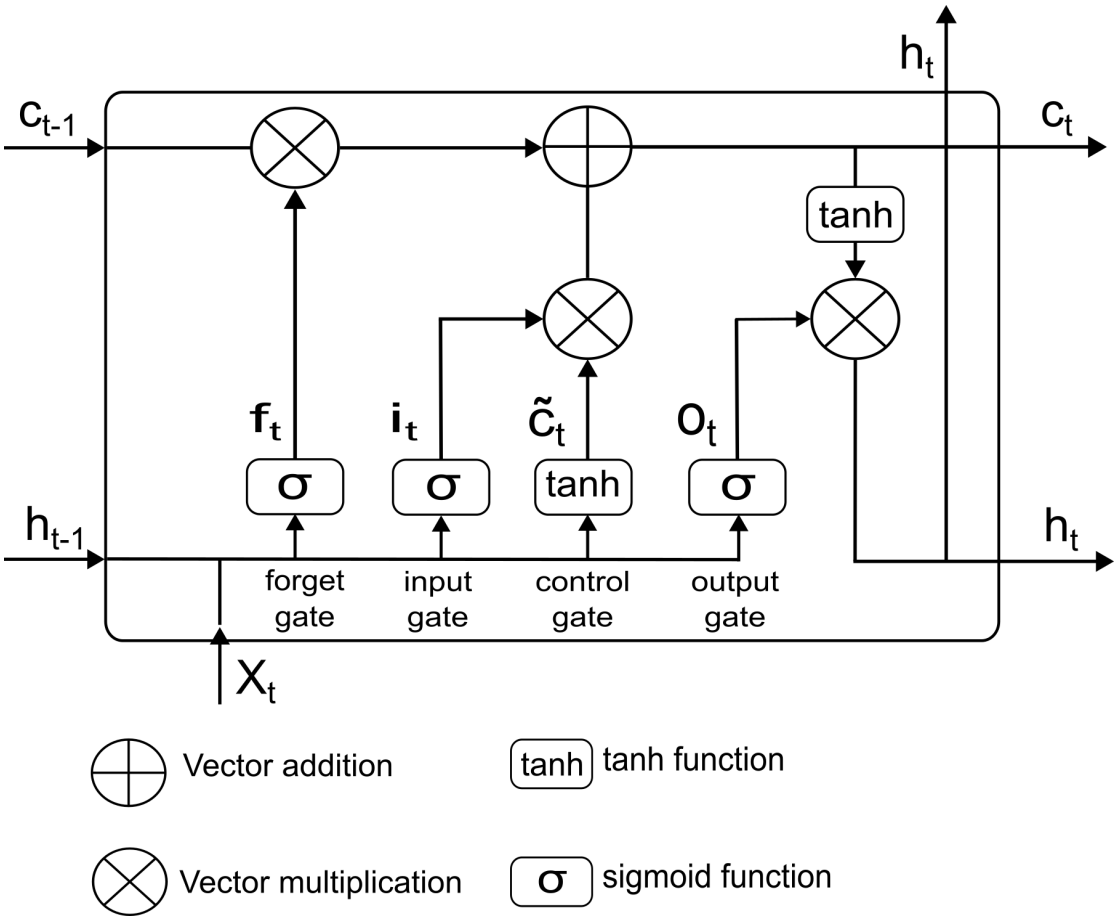


FIGURE 2.2: Architecture of a single LSTM cell adapted from [93]

determines an optimal energy dispatching plan that is calculated based on the day-ahead grid market and a real-time model predictive control (MPC) method that minimises the utilization error and indoor comfort violations caused by following the scheduled energy dispatching plan. In a simulation case of a residential hall community at Cornell University with on-site renewable energy resources, Xiao et al reported improvements in indoor thermal comfort control, ranging between 65.4 and 68.7% and energy cost savings between 29.5 and 39.7% compared to a baseline controller. Kim et al implemented an LSTM-based automatic peer-to-peer (PTP) energy trading model [98]. The model attempts to maximise profits of a virtual three to four-person household in South Korea that generates electricity through a solar PV system. Their energy trading model achieved higher profits every month in both a progressive and fixed rate system throughout the year, and they found that participation in the trading system not only earned monthly profits but also reduced losses from over-generation of electrical energy.

2.8 Research Gap

It has been shown that there is substantial literature detailing the implementation of load forecasting methods to extend and improve the functionality of residential energy management, energy trading and GEB systems. The LSTM is regarded in literature as a neural network architecture well suited to performing time-series forecasts with extensive application in the aforementioned systems.

However, there is a lack of research regarding the real-world time and resource requirements for developing load forecasting models using LSTM. Developing a high-performance LSTM model requires the selection and tuning of various design parameters and data features [95, 96]. This model tuning process is often described as a process of trial-and-error. In practice, a trial-and-error approach could cost time and resources that may make an LSTM modelling implementation unfeasible. In the literature various optimisation techniques are described to tune and select parameter values such as the number of LSTM layers, number of network neurons, dataset features and time lags [8–10]. Yet, the literature has not addressed the practical implications of parameter selection such as the cost of measurement devices to collect training data at a given resolution or the time and compute resources required to achieve longer prediction horizons.

This work aims to address this gap by investigating to what extent the design choices for a residential, LSTM load forecasting model can be inferred. This will be achieved by exploring the relationships between model design parameters and the predictive model performance, and determining if they can inform model design decisions for a practical load forecasting application.

2.9 Chapter Summary

Global energy demand is projected to continue rising, with residential energy consumption accounting for approximately 27% of the total demand.

In response to rising residential demand there exist DSM strategies to shift residential load and curtail peak demand. These strategies include various dynamic

pricing tariffs: TOU, CPP and RTP. In addition, residential DER have seen increased uptake and enable consumers to independently generate energy.

GEBs are envisioned to interact autonomously with the grid to coordinate DER usage with dynamic pricing signals to make energy management decisions that benefit the consumer. Furthermore, the inclusion of load forecasting methods can further augment and optimise GEB energy management functionality.

There exist numerous classical load forecasting techniques, but neural network predictive models like LSTM have been shown to outperform these techniques for forecasting load profile data. Despite the extensive research implementing LSTM models for load forecasting, there is a lack of literature exploring the trade-offs between model performance and the real-world time and resource costs. This work aims to address this gap by investigating to what extent the design choices for a residential, LSTM load forecasting model can be inferred based on the relationships between model design parameters and model fidelity.

CHAPTER 3

Methodology

This chapter details the machine-learning model design and experimental approach for this dissertation. A clarification of term definitions used for this dissertation are presented in section 3.1. In section 3.2 the dataset selection and preparation is discussed. Section 3.3 details the neural network design of the machine-learning model. The experimental setup is detailed in section 3.4. The metrics used to evaluate and analyse the prediction models are described in section 3.5. Finally, the chapter is summarised in section 3.6.

3.1 Declaration of Terms

In this dissertation, "training" data refers to the set of data that is directly applied to the neural network and used to fit the model parameters. The "validation" data is the set of data used to evaluate the developing model during training, and then based on this evaluation tune the model parameters. Lastly, the "test" data is the set of data used to evaluate the performance of the models after the training is complete.

3.2 Dataset Preparation

Developing an LSTM predictive model requires applying numerous samples of time-series, load profile observations to the neural network. This network will use each observation to form a complex system of weighted outputs that can discern latent patterns within the load profile. Using larger datasets provides a greater

range of observations for the neural network to better fit the model on, which generally improves its ability to learn pattern dependencies. The LSTM will ultimately produce a black box model of weighted variables that relates dependencies between one or more load profile features to estimate a future value of the target variable.

3.2.1 Dataset Selection

A publicly available set of residential power measurements, aggregated from 9 different appliances and taken over 3 years and 11 months at a one-minute resolution was chosen as the observation dataset for this work [99]. Training an LSTM model using a dataset which spans a large time frame and contains a wide range of appliances representing various domestic activities enables better model generalisation. The relatively high one-minute resolution ensures that timing information for short and long appliance events are captured. In table 3.1 a summary of the dataset characteristics are shown.

Measurement Duration

The dataset consists of measurements starting from December 2006 and ending in November 2010 for a total of 2,075,259 observations. This is a relatively large dataset compared to similar load forecasting work in which several datasets are in the order of thousands to tens of thousands of observations [8, 100–102]. Having a large volume of observations that also spans a wide time range is generally advantageous for training effective LSTM models [4–7].
Using a larger observation set enables the LSTM neural network to better determine the patterns and relationships between input and output parameters and

Table 3.1: Summary of load profile dataset characteristics.

Duration	Total Observations	Resolution	No. of Appliances
47 Months	2075259	1 Minute	9

their changes over time. In addition, having a greater number of observations during training reduces the risk of producing a model that over fits the data. A model that overfits generalises poorly on new observations i.e., it is able to effectively predict on observations from the training or validation data but less effectively on unseen observations [103].

Measurements taken over longer periods capture more detailed observations of behaviours that vary over time. In a domestic setting this can include daily, weekly and annual variations in behaviour such as: morning versus evening routines, weekday versus weekend activities and responses to seasonal changes in weather (e.g., increased space heater usage during winter). By including more observations of time-dependant events, the dataset has a higher degree of representation of actual domestic activity. This characteristic has been shown to improve the ability of subsequent predictive models to generalise on unseen domestic load profile observations [104–108].

Data Resolution

Each dataset observation consists of minute-averaged samples. This is a relatively high resolution compared to similar load forecasting work in literature which use dataset resolutions ranging from 5-minute to yearly samples [109–112]. Higher measurement resolution ensures that the timing information for each usage event is correctly recorded [113]. For a 1-minute resolution this enables detailed measurement of short-duration appliance usage e.g., electric kettle operation times range from 2-8 minutes [114, 115].

Using a the relatively large minutely resolution enables a wider range of downsampled resolution options. Downsampling is preferable to upsampling which introduces interpolated observations into the dataset. The ability to resample data can be useful for experimental purposes as different models can be trained and tested on datasets of various resolutions.

Appliance Variety

To produce models that are able to more effectively generalise on a domestic load profile it is important to use training data that includes observations representative of typical domestic activities [104–108].
The main domestic appliance use-cases are for cooking, washing, lighting and space heating/cooling [116–118]. This dataset includes measurements from a range of appliances that are representative of these domestic activities: cooking (electric oven, microwave, refrigerator), washing(dishwasher, washing machine, tumble-drier, electric water heater), space heating/cooling(air conditioning) and lighting. Including observations from a wider range of appliances will aid model generalisation capabilities by generating training data that contains more diverse power usage patterns.

Load Profile Variation

Residential energy consumption is influenced by various seasonal factors such as weather changes, household occupancy and appliance usage [119–121]. As a consequence, the load profile dataset will contain variations in the power usages that form seasonal patterns. In figure 3.1 a box-plot analysis of the household power measurements over 3 years illustrates the occurrence of seasonal power usage variation. It can be seen that the smallest values of minimum, median and maximum power usages occur during the summer seasons. As summer changes into fall the power usage increases. The power usage values reach a peak during the winter season of each year. As winter changes to spring the min, median and maximum power values decrease to values similar to the fall season of the same year. In table 3.2 a summary of the box plot values for each year and its respective seasons are shown.

Across each year the median and maximum power usage is lowest during the summer season with peak values of 0.31kW and 2.78kW respectively, observed during the summer of year 1. As summer transitions into fall the median and maximum power usage show average increases of 112.82% and 48.14% respectively across the

TABLE 3.2: Seasonal minimum, median and maximum power usage for each year of the training dataset.

	Yr. 1 (2007-2008)			Yr. 2 (2008-2009)			Yr. 3 (2009-2010)		
Season	Min (kW)	Median (kW)	Max (kW)	Min (kW)	Median (kW)	Max (kW)	Min (kW)	Median (kW)	Max (kW)
Summer (Jun-Aug)	0.08	0.30	2.78	0.08	0.31	2.08	0.12	0.35	2.30
Fall (Sept-Nov)	0.12	0.52	3.48	0.10	0.76	3.49	0.16	0.77	3.52
Winter (Dec-Feb)	0.19	1.32	4.34	0.20	1.30	3.86	0.22	1.38	4.15
Spring (Mar-May)	0.13	0.64	3.35	0.09	0.77	3.28	0.15	0.79	3.27

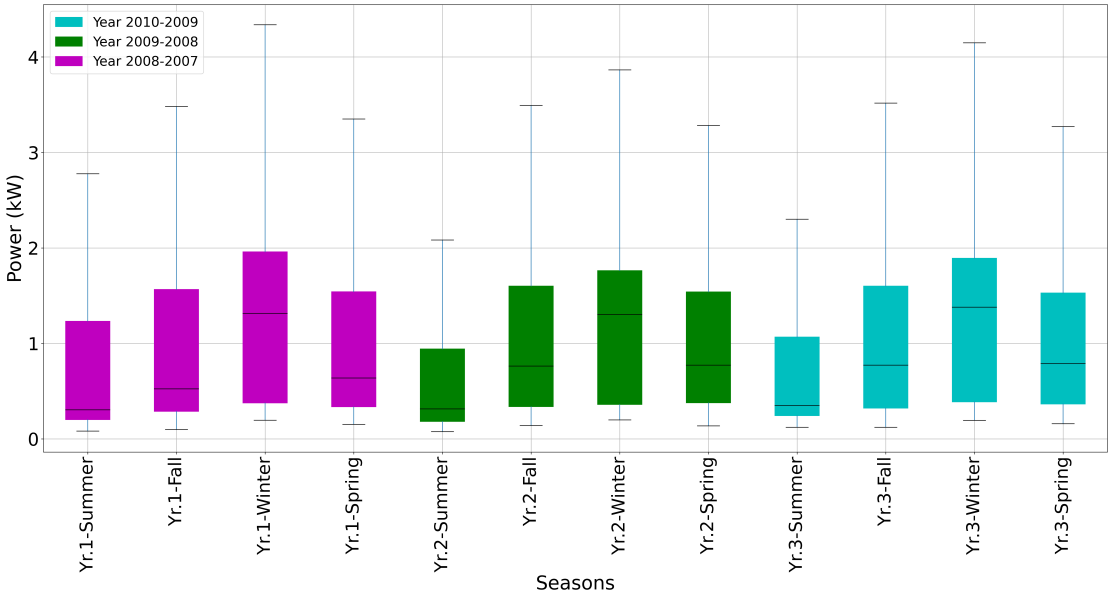


FIGURE 3.1: Box-plot analysis of seasonal power usage across 3 years.

3 year period. The largest values of min, median and maximum power usage are observed during the winter seasons of each year, with further average increases of 101.38% and 17.94% respectively. Finally, as winter transitions into spring the median and maximum power usage decreases by an average of 45.01% and 19.86% respectively.

The seasonal variation present in the load profile data can make model generalisation more difficult, but to perform effective load forecasting it is a factor that must be considered during model development.

3.2.2 Data Cleaning

Approximately 1.25% of the observations within the dataset consist of missing measurements. Observations with missing measurements cannot provide meaningful information about the load profile and are not useful for predictive model training.

To correct for this each missing observation is substituted with an approximation of their true value. The deterministic hot-deck imputation method was chosen to perform these approximations [122]. Hot-deck imputation is a widely used, computationally simple and effective way to handle missing data by replacing them with other dataset observations as approximations of the true values of the missing measurements [122–125]. For this work, hot-deck imputation is achieved by replacing each missing measurement with the corresponding observation from the previous 24-hours.

3.2.3 Normalisation

The relative scale of values in a training dataset can introduce undesired bias into machine-learning models [126–128]. The neural network can perceive features with larger values as more significant than features with smaller values because of the differences in size. This difference in scale can also impact model training duration by increasing the time it takes for the neural network weight optimisation methods to converge [129–131].

To remove the scaling bias and improve the weight optimisation convergence rate it is necessary to normalise the feature values [132–135]. This requires rescaling the values so that they all fit within the same range. For this work, the power values are normalised to fit within a range of [0,1] using the min-max scaling method [136].

Applying normalisation prevents scaling bias by fitting features within a common range of values. No single feature is predominantly larger than any other. Removing disproportionate differences in scale ensures that each feature contributes more

equally to the model fitting process which improves model performance [132, 134]. Furthermore reducing the range of values enables the neural network weight optimisation functions to converge on solutions faster, thereby reducing model training times [137].

3.2.4 Feature Selection And Encoding

To produce an effective prediction model it is important for the training data to consist of particular subsets of the dataset (i.e., features) which are most likely to be strong predictors of the output variables that are to be predicted [138–140]. Including irrelevant features as part of the training data can negatively impact the model performance [141–143].

The dataset chosen for this work measures nine different variables including date-time, active and reactive power and sub-metered measurements from three rooms in the household. For this investigation only four features were used: aggregate power, time of day, day of the week, and day of the year. The power values were taken directly from the dataset measurements. Each of the time based features were derived using the dataset timestamps.

The measured power is a suitable feature because it is the target variable of the predictive models. Historical data is shown to be a strong indicator of future behaviour and consequently a good predictor [144–146]. Time-based features are used to inform when an observation is expected to occur and the timing of cyclical events. Residential energy usage is strongly influenced by changes in time and often form daily, weekly and annual usage patterns [147–149].

The measured power and time stamps were identified as the minimum appropriate features to train a performant load prediction model while also accounting for seasonality. While additional household measurements such as voltage and disaggregate power or exogenous features such as weather could potentially improve model performance they also introduce additional compute cost and complexity to the model design. This study investigates LSTM requirements for residential metered power, which is an aggregated value. Furthermore the inclusion of more

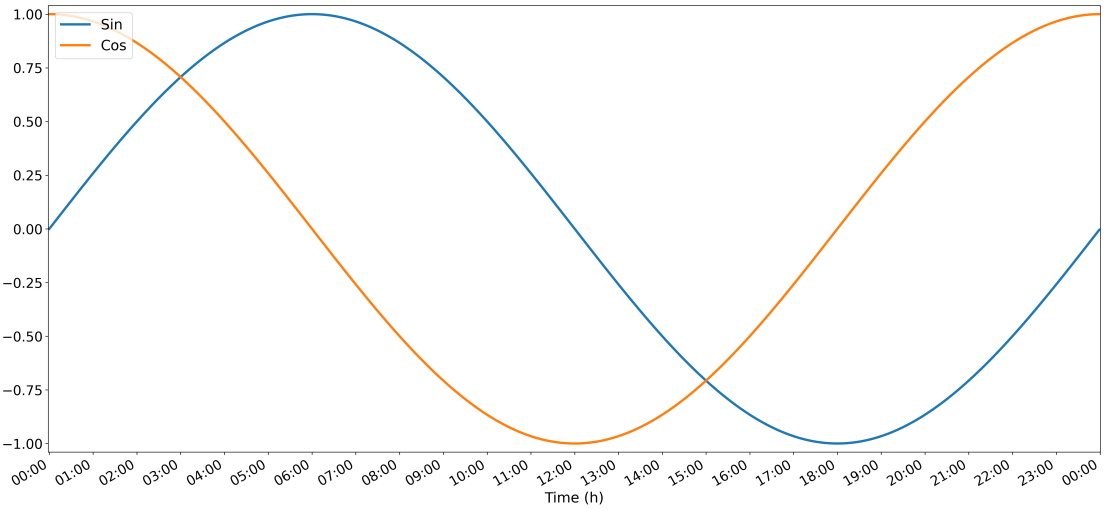


FIGURE 3.2: Encoding of timestamp labels over a 24-hour period into a numerical signal using sine and cosine transforms.

features does not necessarily improve the ability to answer the stated research question and could actually obfuscate the relationship between changing parameters and model performance i.e., are changes in performance attributed to the different data parameters or variations in multiple features over time?

To form a training dataset suitable for an LSTM network requires every feature to be expressed as a numerical value. The power feature is represented numerically in the dataset by default, but the time-based features are represented as text using timestamps.

To express the time-based features as numerical values a sine-cosine transform was chosen to encode the timestamps for each time period (i.e., daily, weekly and annually) as unique pairs of values $(\hat{t}_{sin}, \hat{t}_{cos})$:

$$\hat{t}_{sin} = \sin(\frac{2t\pi}{T}) \tag{3.1}$$

$$\hat{t}_{cos} = \cos(\frac{2t\pi}{T}) \tag{3.2}$$

Where t is the timestamp expressed in Unix epoch seconds and T is the time period expressed in seconds.

Using this transform ensures unique encodings for each timestamp value while also retaining ordinality and relative distances between each point in time. Figure 3.2

shows the sin-cosine encoding for timestamp features over a 24-hour period. Each point in time corresponds to a unique pair of sine and cosine values. Because sine and cosine are periodic functions the ordinality and relative distance between data points in time are maintained.

3.2.5 Training and Validation Data Selection

As discussed in section 3.2.1 residential energy consumption is strongly influenced by seasonal changes. This results in the formation of power usage patterns that vary with each season (e.g, increased usage of electric heaters in winter). Additionally, there are stochastic variables within a household that change over time as a consequence of day-to-day living that can also influence how energy is consumed (e.g., number and age of residents, lifestyle variation, shifting appliance trends).
An LSTM can be used to fit a predictive model on arbitrarily selected datasets, but the model performance can be impacted impacted by the variation within the data points used to produce it. The dataset consists of measurements that reflect real-world power usage which can change stochastically. The frequency, timing and magnitude of usage forms patterns that can shift over time. The aim of the LSTM neural network is to optimise a predictive model based on a set of patterns learned from the training dataset. Changes in the patterns within the training data can diminish model performance as the network weights have not yet adjusted to fit the new usage behaviour. Model performance can be improved if the network is provided additional training data samples of the changing pattern behaviour. This will enable the network weights to fit the model to the changed pattern.
To prevent abrupt seasonal pattern changes within the training data and ensure sufficient data samples of each season, the load profile dataset is partitioned into three separate and contiguous sets of data. Each set consists of one year (365 days) of power measurements. Figure 3.3 illustrates this dataset partition. The training portion will be used to fit the LSTM model and the validation portion will be used to evaluate the model performance during development.

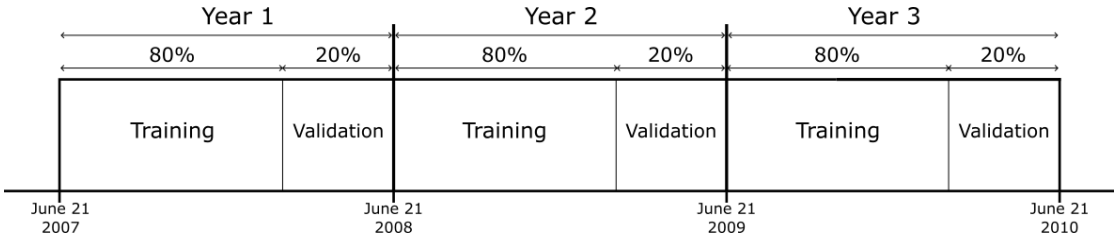


FIGURE 3.3: Partitioning of dataset measurements into three contiguous training data subsets.

Each subset of measurements begins on June 21 to coincide with the start of the summer solstice as observed in France and ends at the start of the summer solstice the following year. This ensures that each subset begins approximately at the start of a season and ends approximately at the start of the next cycle for the same season. The widely used 80% training, 20% validation ratio was chosen to select the portion of data points to be used for training and validation respectively [150–152].

Partitioning the data in this manner achieves three things. Firstly, it ensures that the LSTM is trained on data points that will include power usage variations specific to each year's seasons i.e, summer, fall, winter and spring. Secondly, the training and validation data points will not abruptly start or end in the middle of a season, ensuring that the LSTM is trained on the entirety of each season's power usage patterns. Lastly, implementing the training-validation split enables the model development to be monitored for indicators of overfitting and corrected. These factors will improve the ability of the predictive models to generalise on unseen data as they are more robust against variations in power usage due to seasonal changes.

3.2.6 Test Data Selection

A test dataset is used to evaluate the predictive model performance. Model predictions are performed using features from the test data as inputs and the resulting predictions are compared against the observed test data values.

For this investigation several different predictive models are trained using various

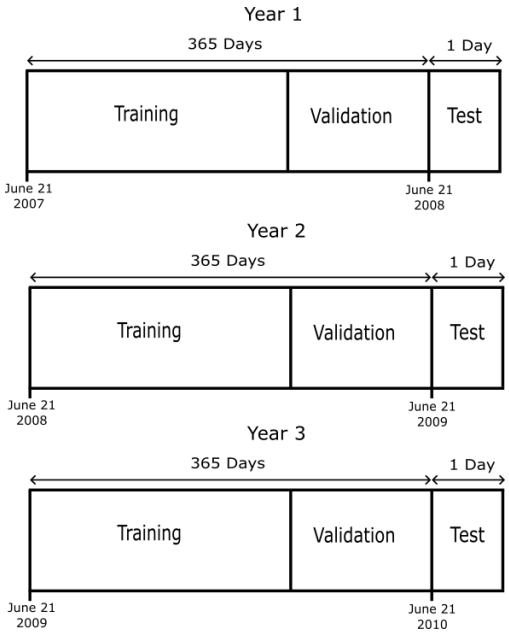


FIGURE 3.4: Test data allocations for the training-validation subsets of each year

parameter settings. To ensure fair performance comparisons each model is evaluated using the same test datasets.

The actual test data is an unseen set of load profile measurements i.e., does not include any data used for model training or validation. The 24-hour period of measurements immediately after the end of each training-validation subset in section 3.2.5 were chosen as the test datasets. Figure 3.4 illustrates the test data selection for each of the 3 subsets. Models trained on a given subset will be evaluated with that subset's test data e.g., models developed using the Year 1 training-validation subset are evaluated on the Year 1 test data. Choosing the immediate 24 hours ensures that the test data is continuous with the training-validation data. This reduces the risk of the test data introducing completely new usage patterns that may occur over time and that the neural network has not yet been able to fit.

3.3 Neural Network Design

The LSTM neural network was designed to overcome the vanishing gradient problem experienced by standard RNNs. The LSTM is able to retain long and short-term dependencies within time-series data patterns. This means it can make use

of pattern information that occurs over a long period of time as well as shorter, more recent changes when determining the neural network weighting. The ability to retain relevant pattern information over extended periods of time makes the LSTM well suited to handling sequences with long-term dependencies, where earlier time steps can have an impact on the outcome of future time-steps.
This section details the LSTM neural network design. Section 3.3.1 describes the system specifications of the machine used to develop and train the LSTM and predictive models. In section 3.3.2 the LSTM architecture and design choices are explained.

3.3.1 System Specifications

Model development and training was performed using a desktop PC with a 3.70 GHz AMD Ryzen 5 5600X 6-Core processor with 16GB RAM, and a GeForce RTX 3070Ti graphics card with 8GB RAM.
The Tensorflow development platform with Keras API was used to build the LSTM architecture and train the models [153, 154]. Tensorflow offers a large repository of open-source machine-learning libraries that supports CNN architectures. The Keras API simplifies the development process through high-level interfaces and facilitates the usage of the CPU and GPU for faster model training.

3.3.2 LSTM Architecture

An overview of the LSTM neural network used for this investigations is shown in figure 3.5. A stacked LSTM architecture composed of three sequential neuron layers was chosen for this work. In a stacked architecture the model parameters are distributed over the entire model space without increasing memory capacity, which enables the model to accelerate convergence and improves non-linear operations on the training data [155, 156]. The first layer consists of 100 neurons, while the second and third layers contain 50 neurons each. The outputs of the final LSTM layer are sent to a linear-output feed forward layer which maps the

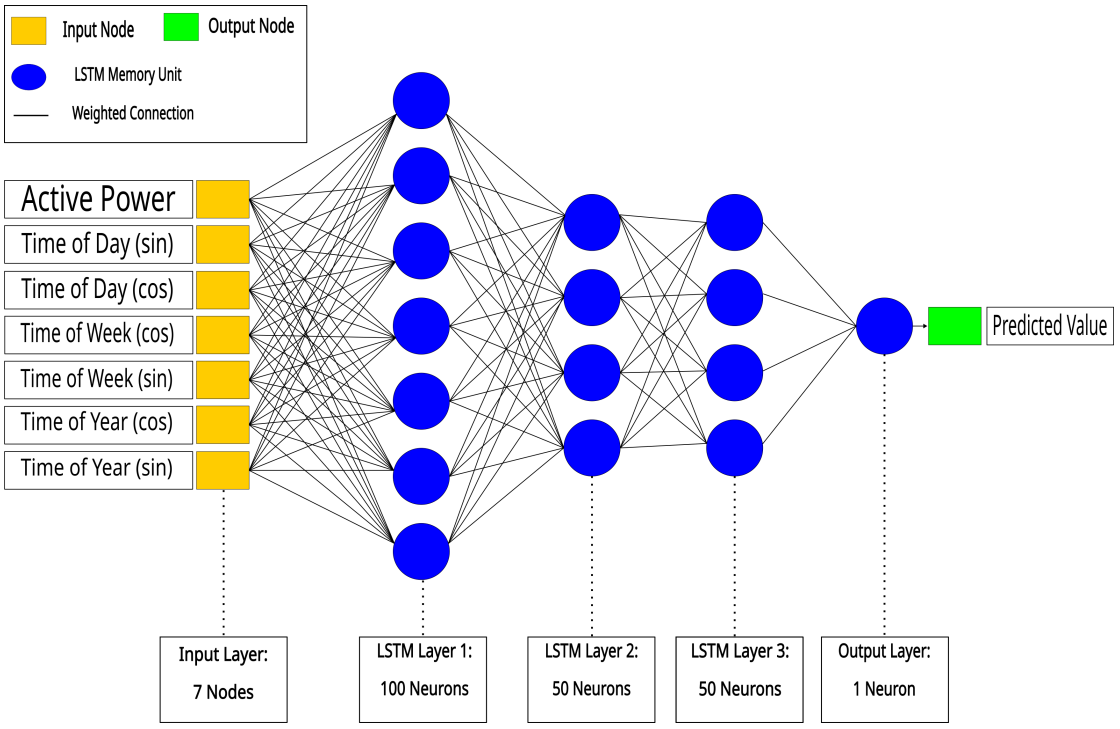


FIGURE 3.5: Outline of the stacked-LSTM neural network architecture

intermediate LSTM layer outputs to a single value i.e., the power prediction for the target time interval.

For this investigation the hyperbolic tangent (tanh) activation function, mean absolute error (MAE) loss function and adaptive moment estimation (Adam) optimiser with a 0.001 learning rate were chosen as the LSTM hyperparameters. Unlike the design parameters (i.e, training data size, data resolution and prediction horizon length) which are varied as part of the investigation, the hyperparameters remain fixed for each experiment.

Activation Function

The tanh was chosen as the activation function. The tanh function has been shown to be effective in LSTM network for modelling patterns with non-linear relationships [157–159].

Loss Function

The MAE was chosen as the loss function. The loss function is used to estimate the error of each model iteration during training so that the neural network weights can be updated to reduce the loss on the next model iteration. The MAE is a linear score that determines the average absolute error between the observed and model predicted values and is formulated as:

$$MAE = \frac{\sum_{i=1}^{N} |\hat{a}_i - a_i|}{N} \tag{3.3}$$

Where $\hat{a}_i$ is the model predicted value, a_i is the observed value and N is the total number of training data samples.
The MAE assigns an equal weight to all errors regardless of their magnitude which makes it more robust against outliers in the dataset: outlier values do not strongly influence the MAE calculation.

Training Epochs

A training cycle of three epochs was chosen. For this LSTM design epochs greater than three were found to produce negligible improvements in model performance.

Gradient Descent Optimiser

The Adam algorithm with a learning rate of 0.001 was chosen to perform the network weight optimisations for this LSTM [160]. This optimiser is widely used in ML applications due to its computational efficiency, fast convergence rate, suitability on non-convex optimisation problems and has been shown to perform well on neural networks using a 0.001 learning rate [160–166]. Another widely used optimiser algorithm, stochastic gradient descent (SGD), is regarded as having better generalisation capabilities than Adam [167]. However in the case of performing electrical load forecasts using LSTM, the generalisation of Adam has been shown to

outperform SGD [162, 168, 169]. Furthermore SGD requires tuning of the learning rate which can result in slower convergence rates and increased complexity, especially when used with more complex networks such as LSTM [170].

Regularization

Dropout regularization is applied to each LSTM layer to reduce overfitting. This method probabilistically excludes the input and recurrent neuron connections to LSTM units from activation and weight updates during model training [171, 172]. By dropping these connections the model becomes less sensitive to the specific weights of individual neurons i.e, reduces bias towards specific patterns in the observations. Performant dropout rates have been shown to range from 0.2 to 0.5, and so a dropout rate of 0.2 was chosen [173–175].

3.4 Experimental Setup

Three experiments were designed to investigate the impact of specific design parameter variations on LSTM prediction model performance. In Experiment A models are developed using increasing amounts of training data. In Experiment B the prediction horizon length and training data amount are varied. In Experiment C the sample rate of the training data is varied along with the training data amount.

3.4.1 Experiment A: Training Data Amount

In Experiment A several power prediction models are produced using sets of increasingly large training data. The computational resource cost for each model is evaluated by measuring the duration of each model training period.
It is generally understood that applying additional training data to an LSTM network improves the performance of the resultant prediction model [4, 140, 141]. This is because a larger training dataset provides more data points for the LSTM

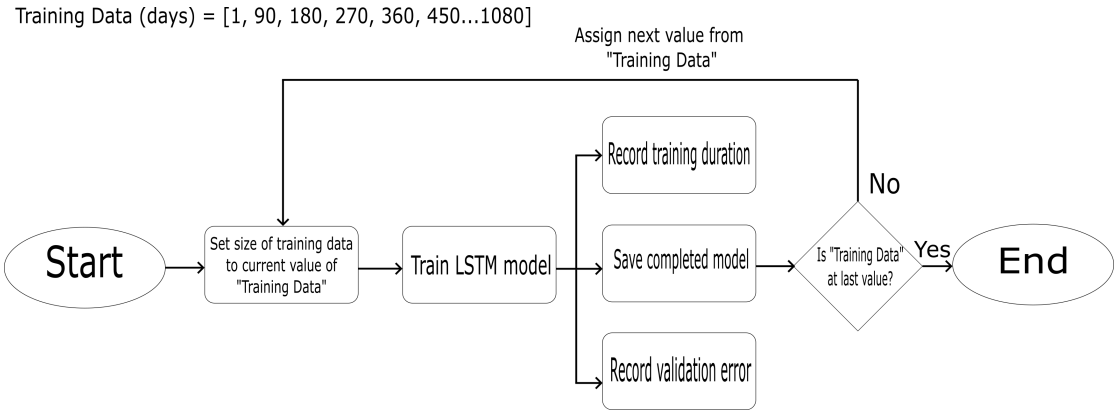


FIGURE 3.6: Flow diagram of Experiment A algorithm

to more accurately adjust the neural network weights to the profile pattern and improve the model's estimation capabilities.

This experiment aims to observe the extent to which increasing amounts of training data can improve model prediction accuracy and the computational cost required. Figure 3.6 outlines the algorithm used to perform this experiment.

First, values for the size of the training data parameter are chosen. Starting from 1 day the training-validation data amount is increased in 90-day increments. This is done so that each training data set encompasses approximately one complete season (i.e., summer, fall winter, spring) of load profile patterns. This is done to minimise the impact of seasonal changes in the training data measurements on the prediction performance of each model.

Next, starting with the 1-day training-validation set a prediction model is trained and the duration for training is recorded. After three epochs the model training is complete. The training duration and model information are stored and the training-validation data amount is updated to the next size increment. This process is repeated until a model has been produced for each of the thirteen training-validation data amounts.

Lastly, each model is evaluated by performing predictions using the 24-hour test data sets set of test data measurements. The MAE between the predicted and observed values for each model is compared and plotted. The results for this experiment are discussed in section 4.1.

3.4.2 Experiment B: Prediction Horizon and Training Data Duration

In Experiment B several power prediction models are produced using various prediction horizon lengths. The computational resource cost for each model is evaluated by measuring the duration of each model training period.

The prediction horizon parameter determines the number of time steps ahead that the model will predict. Forecasts that use a prediction horizon length greater than 1 are known as multi-step forecasts [176]. Multi-step forecasts can be challenging to perform as each additional time step increases the prediction uncertainty as the number of output variables that neural network must simultaneously estimate increases [176–178].

It is generally accepted that applying larger amounts of training data to an LSTM network can improve the performance of the resulting prediction model [4, 140, 141]. Therefore, this experiment aims to observe the extent to which an increase in training data amount can improve the model accuracy for increasingly large prediction horizons and the computational cost required. Figure 3.7 outlines the algorithm used to perform this experiment.

First, values for the size of the training data parameter are chosen. Starting from 1 month the training-validation data amount is increased in three month increments. This is done so that each training-validation data set encompasses

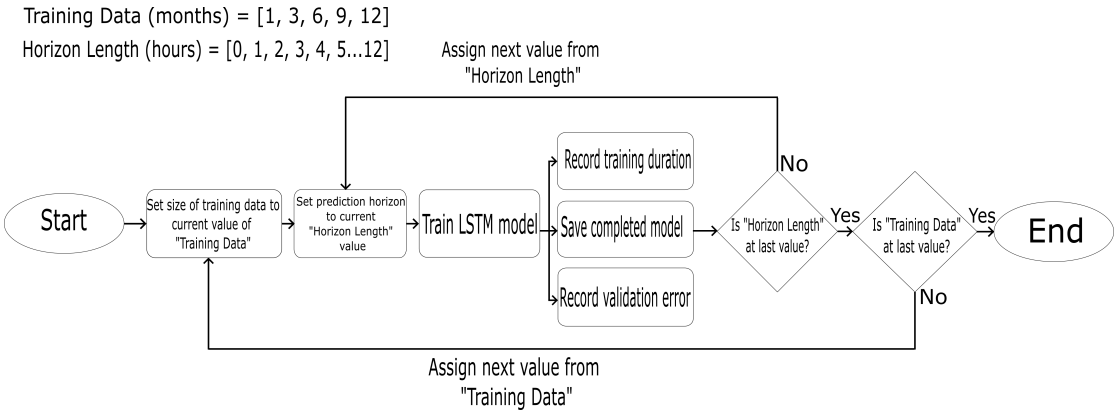


FIGURE 3.7: Flow diagram of Experiment B algorithm

approximately one complete season (i.e., summer, fall winter, spring) of load profile patterns. This is done to minimise the impact of seasonal changes in the training data measurements on the prediction performance of each model.

Next, the prediction horizon parameter values are chosen from a set of thirteen hourly prediction horizon lengths {0.5, 1, 2, 3, 4, 5...12}. These values were chosen based on prediction horizon lengths used in load forecasting application literature [8, 55, 82].

Then, starting with the half-hour horizon length and 1-month training-validation data amount, a prediction model is trained and the training duration is recorded. After three epochs the model training is complete. The training duration and model information are stored and the training-validation data amount is updated to the next size increment. After a model has been produced for each amount of training-validation data, the horizon length is incremented to the next value and the training-validation data amount is set to 1 month. This process is repeated until a model has been produced for each combination of the thirteen horizon lengths and five training-validation data amounts.

Finally, each model is evaluated by performing predictions on the 24-hour set of test data measurements. The MAE between the observed and predicted values for each model is compared and plotted. The results for this experiment are discussed in section 4.2.

3.4.3 Experiment C: Training Data Resolution

In Experiment C, five different resolutions of data are used to create prediction models trained on sets of increasingly large training data. The computational resource cost for each model is evaluated by measuring the duration of the model training period.

The default resolution of the load profile dataset is one minute. By reducing the data resolution there is a loss in the profile pattern detail but the amount of memory required to store the profile data also decreases. Figure 3.8 illustrates the difference in pattern detail between the 1-minute and 30-minute resolutions

for the same load profile. It can be seen that various changes in the frequency and magnitude of the profile power and power peaks that are observable at the 1-minute resolution cannot be observed at the 30-minute resolution.

This experiment aims to investigate the relationship between the change in model

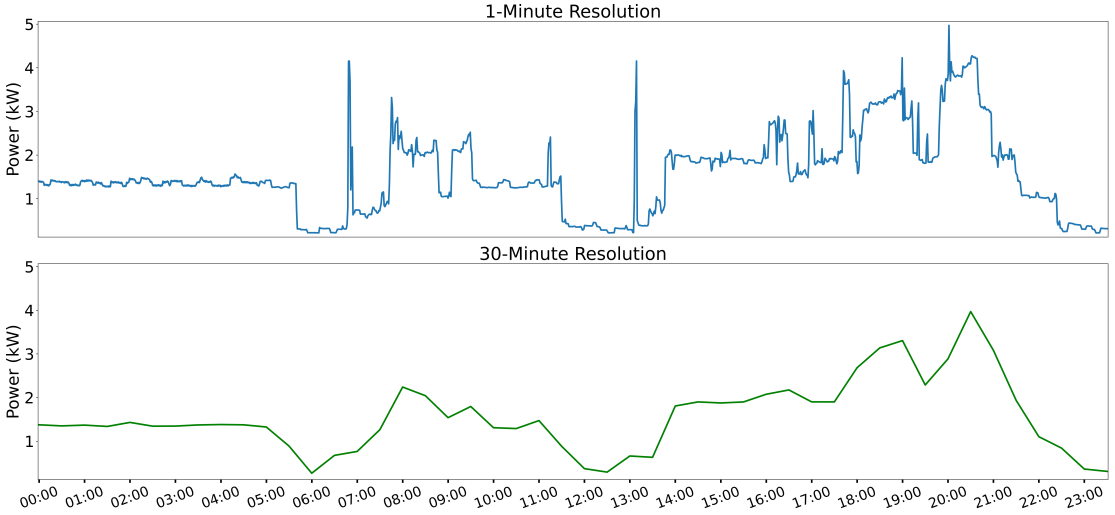


FIGURE 3.8: Comparison of the same set of load profile measurements at 1 and 30-minute resolutions

performance for different data resolutions as the training data amount increases, and the change in computational resources required for training each model. Figure 3.9 outlines the algorithm used to perform Experiment C.

First, values for the size of the training data parameter are chosen. Starting from one month the training data amount is increased in three month increments. This is done so that each training data set encompasses approximately one complete season (i.e., summer, fall winter, spring) of load profile patterns. This is done to

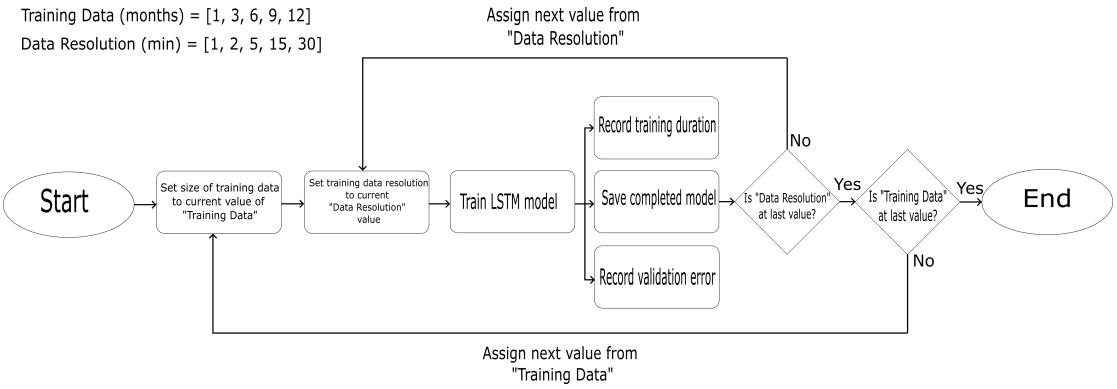


FIGURE 3.9: Flow diagram of the Experiment C algorithm

minimise the impact of seasonal changes in the training data measurements on the prediction performance of each model.

Next the data resolution parameter values are chosen. This set of five different data resolutions includes 1, 2, 5, 15 and 30-minute resolutions. These values were chosen to reflect common resolution settings used to measure power in practical applications [21, 179, 180].

Then, starting with the 1-minute resolution and 1-month training data a prediction model is trained and the training duration is recorded. After three epochs the model training is complete. The training duration and model information are stored and the amount of training data is updated to the next size increment. After a model has been produced for each training data amount, the resolution is incremented to the next value and the training data amount is set to one month. This process is repeated until a model has been produced for each combination of the five data resolutions and five training-validation data amounts.

Finally, each model is evaluated by performing predictions on the 24-hour set of test data measurements. The MAE between the predicted and observed values for each resolution-training size model is compared and plotted. The results for this experiment are discussed in section 4.3

3.4.4 Training and Validation Data Incrementation

Each experiment in this investigation includes incremental additions of training and validation data. As explained in section 3.2.5 the measurements for each of the training-validation subsets begin and end on specific dates. As each experiment progresses the training-validation data is incremented by selecting measurements backwards in time relative to the end-date of each training-validation subset.This process is illustrated in figure 3.10 for the Year 1 (2007-2008) subset. As the required training-validation data size increases the measurements extend further back in time from the reference end-date i.e. June 21 2008.

This approach keeps the training-validation measurements continuous while also remaining contiguous with the test data subset. This means that models for each

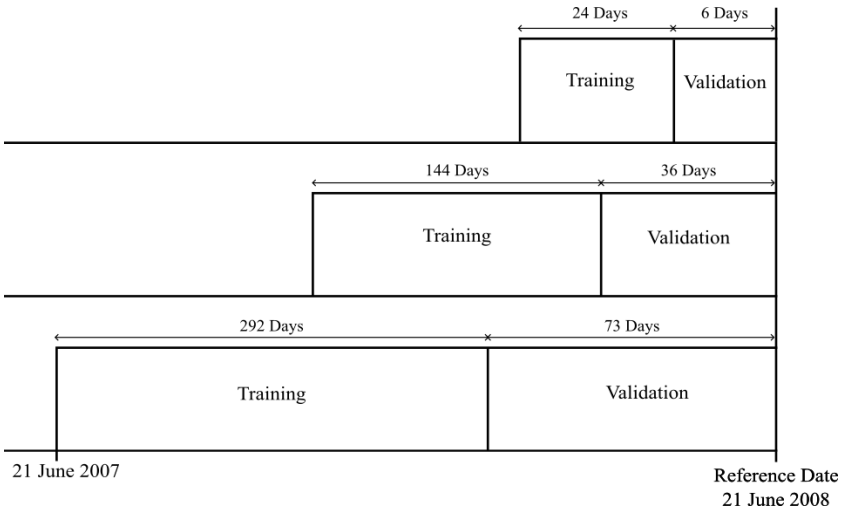


FIGURE 3.10: Selection of measurements for the 1, 6 and 12-month sizes of training-validation data in the Year 1 subset.

training-validation data subset will be trained and evaluated on similar sets of data. This ensures fair comparison of model performance results.

3.5 Evaluation Metrics

Five metrics were chosen to evaluate the performance and quantify traits of the prediction models produced during each experiment.

3.5.1 Prediction Model Error

The MAE is used to evaluate the forecast performance of the LSTM prediction models. The MAE is one of the most widely used metrics for evaluating the performance of machine- learning models [181–185].

An MAE of zero is indicative of a completely accurate prediction, so MAE scores closer to zero indicate fewer prediction errors and better model performance. Compared to quadratic error scores like the root mean square error (RMSE) the MAE is simpler to interpret as the resultant score is in the same units as the observed and predicted data.

3.5.2 Goodness of Fit

The coefficient of determination (R^2) score is used to evaluate the goodness-of-fit between the predicted and observed data points[186–188]. The R^2 score is used as an indicator of how well the predicted values fit the observed data compared to a mean line of the observed values. The R^2 is formulated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(b_i - \hat{b_i})^2}{\sum_{i=1}^{N}(b_i - \bar{b_i})^2} \tag{3.4}$$

Where $\hat{b_i}$ is the predicted value, b_i is the true value, $\bar{b_i}$ is the mean of the observed values and N is the total number of data points.

An R^2 score of one indicates that the predicted values exactly fit the observed values. An R^2 score of zero indicates that the predicted values fit the observed values equally well as the mean line. An R^2 score less than zero indicates that the predicted values are a worse fit to the observed values than the mean line. For a predictive model to fit as well or worse than the mean line indicates very poor or completely inaccurate prediction performance.

3.5.3 Energy Prediction Error

To contextualise and assess the predictive model performance the model errors are also expressed in terms of energy. This work is based on the practical application of predictive models to forecast power in a domestic setting, and so to meaningfully convey the performance it is useful to express the outputs in terms of the units that are being predicted. If a residential load forecast model performs technically well in terms of its error metrics but poorly predicts the actual energy then it is not suitable for the practical environment it is intended for.

To calculate energy the Trapezoidal Rule was chosen to perform the numerical integrations of the power-time graphs. The Trapezoidal Rule has high convergence

speed and is less computationally intensive than other numerical integration techniques such as the Simpson's Rule. It is formulated as follows:

$$Energy = \sum_{n=0}^{N-1} \frac{1}{2}(f_n + f_{n+1})(\Delta x)_n \tag{3.5}$$

Where N is the total number of samples, $f_n + f_{n+1}$ are the power values at samples n and $n+1$ respectively, and Δx is the time step size at sample n.

To quantify the energy prediction performance for each model a residual energy (RE) value is determined. To calculate RE the residual power (R) is first determined by taking the absolute difference between the observed and predicted values of the power-time graph:

$$R[n] = |c[n] - d[n]| \tag{3.6}$$

Where $R[n]$ is the residual power, $c[n]$ is the observed power and $d[n]$ is the predicted power at sample $[n]$.

The RE is then calculated by taking the numerical integral of the residual power graph. Combining equations 3.5 and 3.6:

$$RE = \sum_{n=0}^{N-1} \frac{1}{2}(R_{[n]} + R_{[n+1]})(\Delta x)_n \tag{3.7}$$

Where N is the total number of samples, $R_{[n]} + R_{[n+1]}$ are the residual power values at samples n and $n+1$ respectively, and Δx is the time step size at sample n.

A RE value of zero indicates an exact energy prediction. RE values greater than zero indicate larger amounts of erroneously predicted energy and poorer prediction model performance.

3.5.4 Peak Power Prediction

Peak power demand is an important load profile characteristic for determining energy pricing.

To evaluate the predictive model peak power forecast the values of the predicted profile peaks are compared to the observed profile peaks. A mean peak error

(MPE) value is determined by taking the mean of the difference between the peak of the observed power profile and the corresponding predicted power value.
The "find peak" algorithm from the SciPy Python library with a threshold value of 1.2kW is chosen to perform the peak detection. Including this power threshold ensures that the detected peaks are indicative of high household power usage [189, 190]. The MPE is formulated as follows:

$$MPE = \frac{\sum_{i=0}^{N} |P_i - O_i}{N} \tag{3.8}$$

Where P_i and O_i are the predicted and observed peaks values respectively at sample i and N is the total number of peak samples.

3.5.5 Training Duration

The model training duration is used as a measure for the degree of computational intensity required to develop each prediction model. The measured training time is relative to the hardware used for this investigation as described in Section 3.3.1.

3.6 Chapter Summary

In this chapter the definition of machine-learning terms as used in this dissertation are clarified. The selection and preparation of load profile data for model training is explained. The design of the LSTM architecture is detailed as well as the organisation of the training data. The three experiments to meet the stated research objects are explained in detail. Finally, the metrics used to analyse and evaluate the developed prediction models from each experiment are discussed.

CHAPTER 4

Results And Discussion

This chapter details and discusses the experimental results of this work. The results of Experiment A are discussed in section 4.1. In section 4.2 the results and analysis of Experiment B are shown. Lastly section 4.3 details the results for Experiment C. The chapter is summarised in section 4.4.

4.1 Experiment A: Training Data Size

In Experiment A the relationship between model prediction accuracy and the training data size is investigated. This section details the experimental results and provides analyses of the regression, power peak and residual energy prediction models.

The results of Experiment A are shown in figure 4.1. The top graph shows a plot of prediction errors expressed as MAE for several models produced using different sizes of training data. The bottom graph shows a plot of the model training times for each of the training data sizes.

Starting from the plot of MAE vs training data size, the maximum MAE value of 0.069 is observed at the 1-day model with a minimum training duration of 0.4 minutes. A larger MAE indicates poorer prediction accuracy and so this model is comparatively the least accurate.

At the 90-day model the first decrease in MAE occurs, reducing by 61% to a value of 0.027 from the 1-day model. This indicates a significant improvement in prediction accuracy. This model required a training duration of 10.1 minutes which is 25 times longer than the 1-day model.

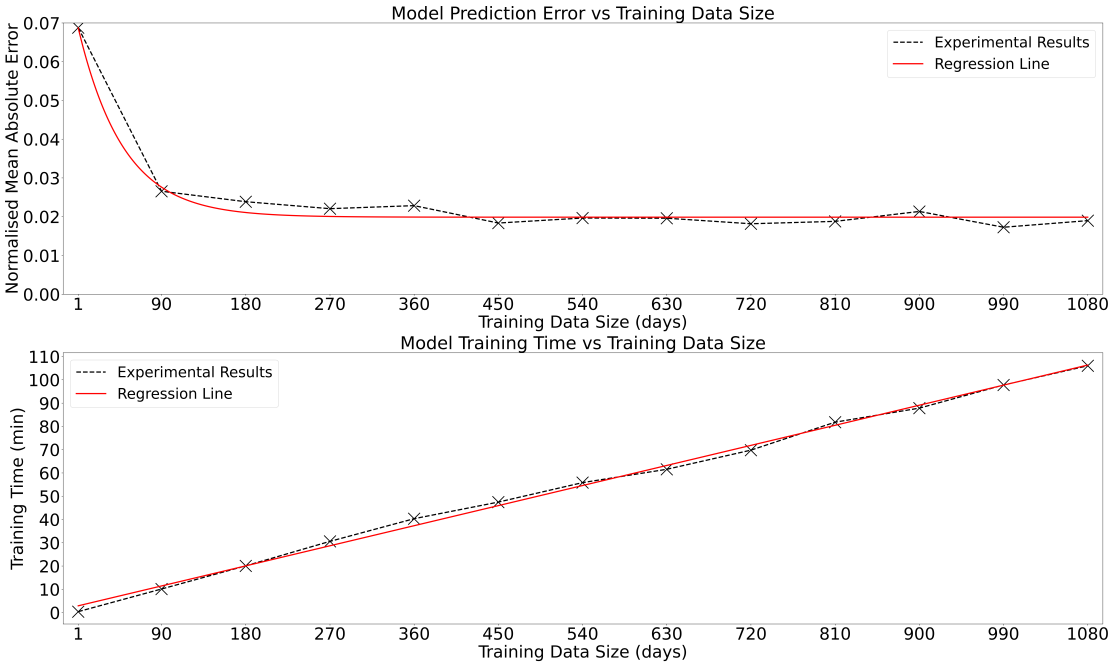


FIGURE 4.1: Comparison of model prediction errors and training time against training data size

As the training size increases beyond 90 days the MAE continues to decrease. At the 450-day model there is a relatively large reduction in MAE to a value of 0.018. The model training duration increases to 47 minutes which is 4.7 times longer than the training period for the 90-day model.

The next appreciable improvement in MAE is observed at the 950-day model. With an MAE of 0.017 this model has the lowest error and highest prediction accuracy. This is a relatively small reduction in error when compared to the 0.018 MAE of the 450-day model. This model's training duration of 98 minutes is also twice that of the 450-day model duration.

A summary of the Experiment A results are shown in table 4.1. These results suggest some relationships exist between the training data size, model accuracy, training duration and model error reduction.

4.1.1 Regression Analysis

To determine the relationships between the MAE, training data size, and model training duration a regression analysis was performed on each of the Experiment

TABLE 4.1: Summary of select Experiment A results.

Training Data Size (days)	MAE	Training Duration (min)
1	0.069	0.4
90	0.027	10.1
450	0.018	47.5
990	0.017	97.7

A results. The regression results are shown in figure 4.1. For the graph of model prediction errors vs training data size an exponential decay curve appears to correspond closely with the observed, experimental results. The equation for this decay curve is formulated as:

$$E = 0.050e^{-0.021D} + 0.020 \tag{4.1}$$

Where E is the model error and D is the training data size expressed in days.
The decay curve fits the experimental results with an R^2 score of 0.98, and its behaviour is consistent with the prediction error trends observed in the Experiment A results: a large reduction in MAE from the 1-day to 90-day model, followed by smaller reductions between the 90-day and 450-day models and then minimal error reductions for models trained beyond 450 days. These indicate that the exponential decay curve is a suitable approximation to model the relationship between the predictive model error and the training data size.
The results of the model training time vs training data size approximate a linear relationship. The linear equation for this graph is formulated as:

$$TD = 0.096D + 2.804 \tag{4.2}$$

Where TD is the training duration expressed in minutes and D is the training data size expressed in days.
The linear graph fits the experimental values with an R^2 score of 0.99, and also exhibits behaviour consistent with the model training time trends observed from

the Experiment A results: linear increase in model training time for increasing training data size.

The linear relationship between training time and data size implies that models trained on larger data will always require more training time i.e., computational resources. This finding together with the non-linear decay of model error with training data size indicates that, for the hardware used to run Experiment A, models trained on progressively larger samples of this load profile data will produce exponentially decreasing improvements in performance while the computational resource cost increases linearly. This means that the improvement in model performance becomes disproportionately small compared to the computational resource cost beyond a certain training data size. For this household that training data size is 450 days. Furthermore, these regression models enable the performance of predictive models for this specific household to be estimated using training data sizes within the range of 1 to 1080 days. The linear and exponential decay functions can be used to interpolate the expected performance of a model for a specific training data size. This enables model performance estimation without using resources to train and evaluate an additional predictive model.

4.1.2 Energy Analysis

To evaluate the energy prediction performance analyses of the predicted and residual energy values were determined for select models produced in Experiment A. A comparison of the power vs. energy predictions and cumulative residual energy over a 24-hour test data period for the 1, 90, 450 and 990-day models are shown in figure 4.2.

The observed total energy over the 24-hour test period is 27.7 kWh. The 1-day model produced the largest energy prediction error of 15.4 kWh. Compared to the observed total energy this represents an error of 55.5%. A poor visual match between the profile predicted by the 1-day model and the observed power profile can be seen. The 950-day model produces a prediction error of 4.6 kWh which represents 16.7% of the observed energy value. Compared to the 1-day model this

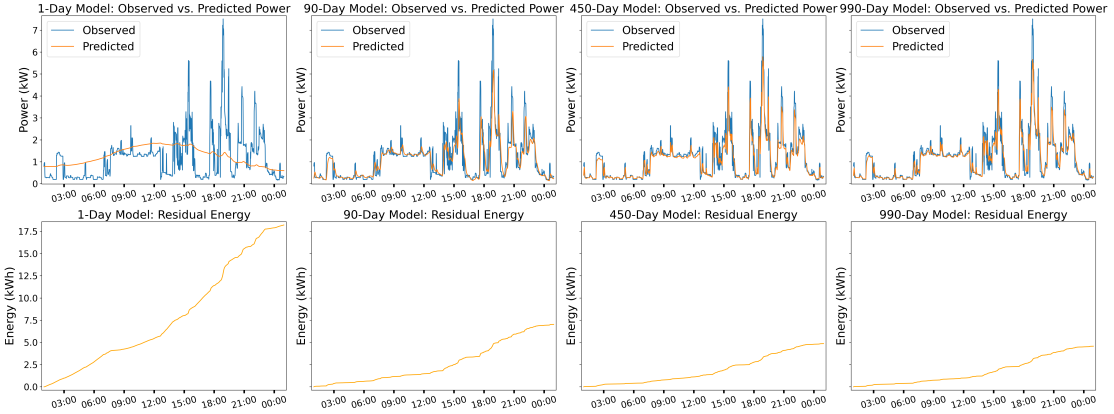


FIGURE 4.2: Comparison of predicted vs actual power and cumulative residual energy for 1, 90, 450 and 990-day prediction models.

is a 38.9% reduction in the residual energy error. There is a close visual match between the predicted power profile of the 950-day model and the observed power profile.

A summary of the energy prediction values for the 1, 90, 450 and 990-day models are shown in table 4.2. As the training data size increases there is a reduction in the residual energy. For larger training data sizes the reduction in residual error decreases.

From these results it is shown that the model energy prediction improves with additional training data size. This finding is similar to that of section 4.1.1 which related the reduction in MAE to an increase in training data size. This suggests that the change in energy prediction accuracy may share a similar exponentially decaying relationship to training data size.

TABLE 4.2: Summary of power and energy predictions for select models.

Training Data Size (days)	Residual Energy (kWh)	MAE (kW)
1	18.22	0.069
90	7.03	0.027
450	4.86	0.018
990	4.57	0.017

4.1.3 Power Peak Analysis

The observed profile and predicted profile peaks from the Experiment A models are compared to evaluate their peak prediction performance. A comparison of the observed power peaks vs. the predicted power profile over a 24-hour test data period for the 1, 90, 450 and 990-day models are shown in figure 4.3. It can be seen that the 1-day model produced the largest MPE, with a value of 1.8 kW. A poor visual match can be observed between the 1-day model predicted power profile and the observed power profile. The 950-day model produces a MPE value of 0.65 kW. Compared to the 1-day model this is a 64% reduction in the MPE. There is a close visual match between the predicted power profile of the 950-day model and the observed power profile. A summary of the MPE for the 1, 90, 450 and 990-day models are shown in table 4.3. As the training data size increases there is a reduction in the residual energy. For larger training data sizes the reduction in MPE decreases.

From these results it is shown that the MPE also improves with additional training

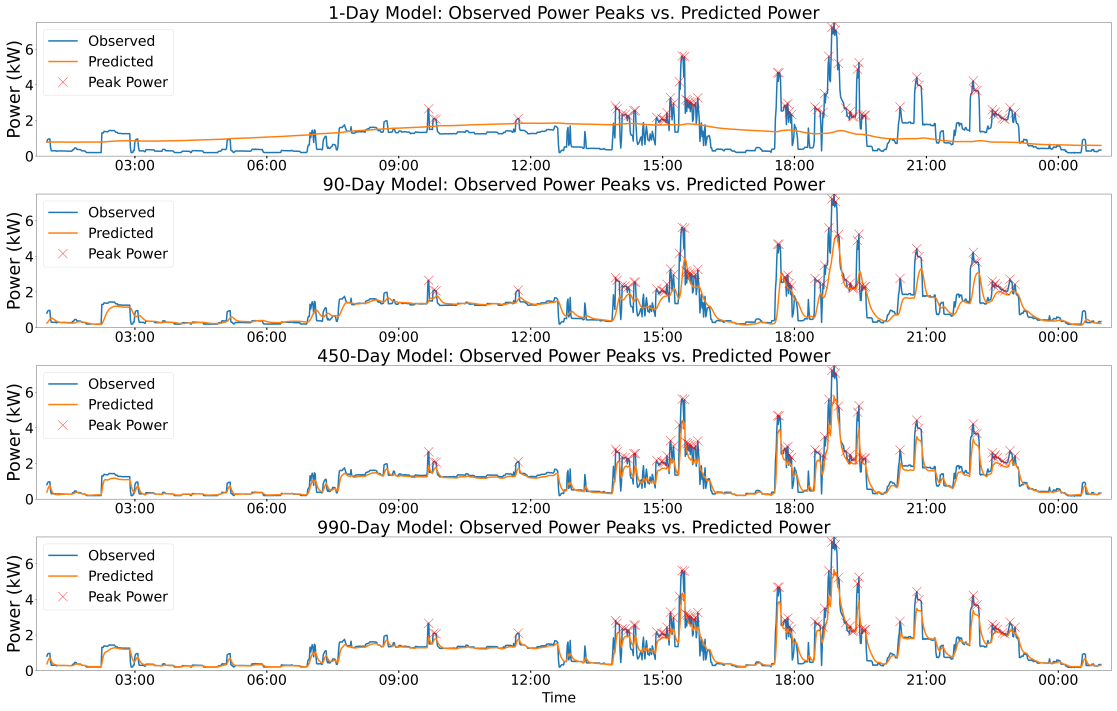


FIGURE 4.3: Comparison of observed power peaks vs. predicted power profiles for the 1, 90, 450 and 990-day models.

TABLE 4.3: Summary of MPE values for select models.

Training Data Size (days)	MPE (kW)
1	1.81
90	1.05
450	0.77
990	0.65

data size, similarly to the previous findings in section 4.1.2.

4.1.4 Prediction Error Variance

From the results of Experiment A it is shown that in general the model prediction error decreases with larger training data sizes. As seen in figure 4.1 this is not the case for every model. An analysis of the statistical variation within the training data was performed to explain the cause of the change in model behaviour. The predictive model performance is dependant on the features provided to the LSTM network during training [138–140]. Each feature data sample is treated as a predictor for the outcome of the variable being predicted and are the basis for updating the weights of the network neurons. Variations within the patterns present in the feature dataset can affect the weighting of the neural network and impact the model's prediction performance.

As discussed in section 3.2 the load profile dataset used for this experiment includes cyclical and stochastic variations in the power usage. A box plot analysis of the training data used during Experiment A is shown in figure 4.4.

At the 360-day model the first rise in MAE is observed. It can be seen that the box plot for the data at this point is statistically distinct compared to the previous 3 data sets which have more similar box plot dimensions. The 360-day box plot has a smaller interquartile range, median and average power compared to the 90, 180 and 270-day models. This suggests that another shift in the power usage pattern has occurred, but because it is dissimilar to the previous 3 patterns the LSTM weights are changing to fit both the recent and previous pattern. As a result there is a slight loss in prediction accuracy.

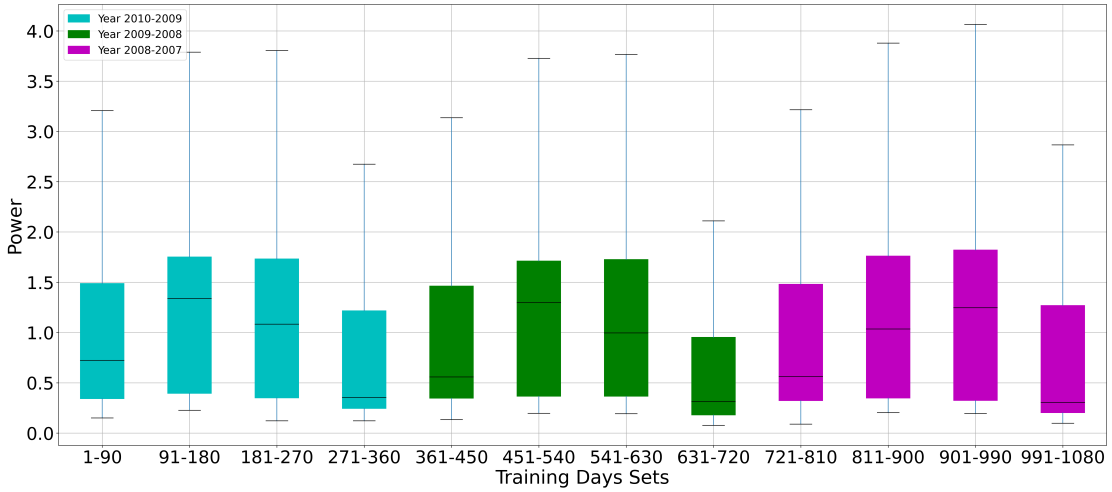


FIGURE 4.4: Box plot analysis of the load profile datasets. Measurements begin in 2010 and extend backwards ending in 2007.

As the training data increases to 450-days a relatively large drop in MAE is observed. At this point the dataset contains a complete seasonal cycle of data points, plus an additional summer cycle. The summer season is relevant as the test data for each of these models contains observations from the summer period. The training data features in the summer season would correlate more strongly as predictors for the test data due to the similarity of their seasonal patterns. With the inclusion of the 450-days training data, two sets of summer season data with similar box plot dimensions have been used to train the LSTM. This is likely to have further improved the predictive ability to estimate load profile events occurring in summer i.e. the test data season.

From the 540 to 720-day box plots, similar box plot dimensions to the previous year are observed. This test data does not provide substantially different pattern information to the LSTM network. This is evidenced by the prediction performance not noticeably improving or worsening.

From the 810-day model onwards the box plot distributions of the 2007-2008 period demonstrate distinct differences compared to similar periods in the 2008-2009 and 2009-2010 years. This corresponds to the fluctuations in model prediction error observed at the 900 and 990-day models.

The analysis of 4.4 indicates that the training data variation has some impact on the model prediction accuracy. Examination of the load profile variance can

be used to estimate the impact of training data sections on the prediction model accuracy.

4.1.5 Discussion

The exponential decay and linear models from the results in section 4.1.1 show that there is a trade-off between the amount of training data, model prediction accuracy and the computational cost: increased training data improves model accuracy for additional training time. From the literature in section 2.7 it is well established that prediction model performance benefits from the application of additional training data, but these models show that the computational cost determines the extent to which this is true.

For a GEBs application, 450 days would be the practical extent to which additional training data would be required to develop forecast models for this particular household and LSTM. This is because, as shown in figure 4.1, beyond this training data size the improvements in model performance become disproportionate to the increase in computational resource cost. This means increased processing requirements and energy costs for the consumer, to produce models with diminishing improvements in accuracy.

This finding is supported by the results of the energy and power peak analysis in sections 4.1.2 and 4.1.3 which show that the improvements in energy prediction and peak detection diminish after approximately 450 training days. This means that changes in the training data produce appreciable changes in the ability to predict energy and profile peaks, and that TOU and energy trading applications for this household, which benefit from energy and power peak predictions, can be improved to a finite extent by increasing training data size.

Using interpolation and the training data size, the exponential decay model can estimate the accuracy of additional predictive models specific to this household. The ability to estimate predictive model performance using only the training data would be useful as an analysis tool to determine the suitability of a load profile

for forecasting applications. It would enable forecast-suitable GEBs to be identified based on an estimation of predictive model performance from their currently available load profile information.

Lastly, the box plot analysis of the training data variation in section 4.1.4 shows that statistical variance in the training data pattern can impact the prediction model performance. For GEBs, which leverage smart metering capabilities to generate load profile data, this means that the quality of their data is not only dependant on the amount but also the power consumption routines of the occupants. This suggests that for a load forecasting application the GEB home with more consistent usage patterns is likely to produce more accurate load models for the same amount of training data.

4.1.6 Summary of Findings

From the results of Experiment A it was found that the relationship between the model prediction error and compute intensity with respect to the training data size can be modelled by an exponential decay and linear function respectively. These models enabled the identification of an upper training data size limit at 450 days; beyond this size the improvements in model performance become disproportionately small compared to the required compute intensity. Furthermore, the exponential decay model enables the estimation of the model performance for an arbitrary training data size within a range of 1080 training days.

The statistical dispersion of measurements within the load profile data was shown to impact the model performance.

It was found that improvements in the model power prediction accuracy produce appreciable improvements in the energy and power peak prediction performance.

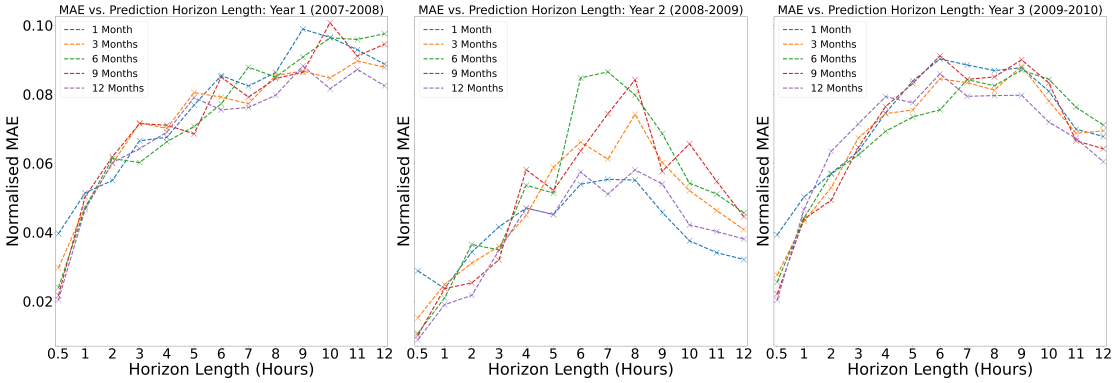


FIGURE 4.5: Comparison of model prediction errors and training data size for various prediction horizon lengths

4.2 Experiment B: Prediction Horizon and Training Data Duration

Experiment B investigates the extent to which an increase in training data amount can improve the model accuracy for increasingly large prediction horizons. This section details the experimental results and provides analyses of the model prediction error, training duration and goodness of fit, as well as the residual energy and power peak prediction performance. A comparison of model prediction errors against the training data size for different prediction horizon lengths is shown in figure 4.5.

Starting from the 1-minute prediction horizon of the Year 1 graph it can be observed that MAE is at a maximum value of 0.040 for the 1-month training model. For the 1-minute horizon length the MAE is seen to decrease for each increase in training data size, reaching a maximum value of 0.020 at the 12-month model.

As the prediction horizon increases there is a general increase in the MAE for models of each training data size. Yet, the rate of MAE change for each model is inconsistent. The models which demonstrate the minimum and maximum MAEs for a given horizon length appear to change independently of the training data size. For instance, at the 5-hour horizon the minimum and maximum MAEs are from the 3 and 9-month models respectively. In contrast, at the following 6-hour horizon the minimum and maximum MAEs are instead from the 1 and 12-month models respectively. At the 10-hour horizon a peak MAE value of 0.1 is observed

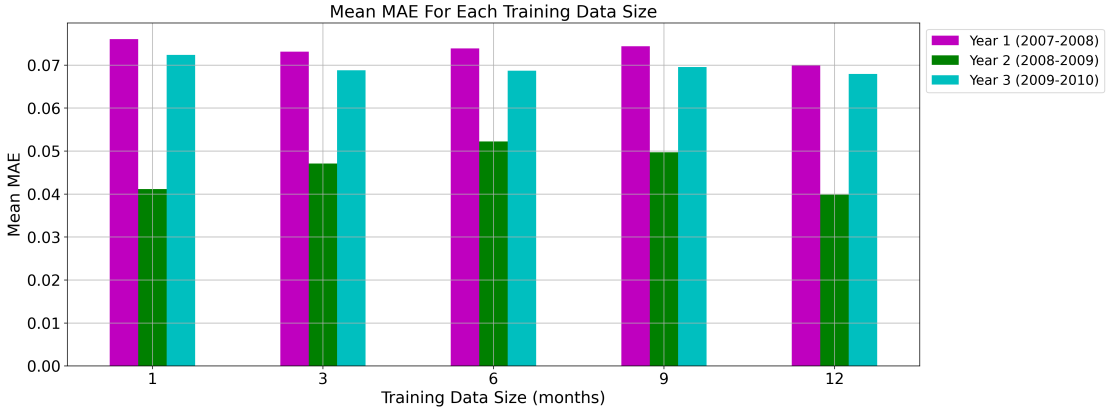


FIGURE 4.6: Comparison of mean MAEs for models of each training data size.

at the 9-month model. For prediction horizons larger than this length, the MAE of several models begins to decrease including the 1, 3 and 12-month models.

The trends in Year 2 are largely similar to those observed in Year 1. The lowest model errors are again observed at the 1-minute horizon length with minimum and maximum values of 0.029 and 0.009 at the 1 and 12-month models respectively. As the horizon length increases, similar inconsistencies in the change of MAE are observed. Unlike the previous year, there is a sharp rise in MAE from the 5 to 7-hour horizons demonstrated by the 67% and 42% increase for the 3, and 9-month models respectively. After the 8-hour horizon there is a general decrease in MAE across each model training size.

The trends and behaviours in MAE and horizon length for Year 3 are similar to those observed in Year 1 and Year 2: minimum MAE values at the 1-minute horizon and increasing MAE for increasing prediction horizon lengths.

4.2.1 Model Prediction Error Analysis

An analysis of the mean MAE for each of the prediction models in Experiment B is shown in figure 4.16. It can be seen that each year the average MAE is similar for models of different training data size.

The results from 4.5 and 4.16 show that the change in MAE with respect to the horizon length varies similarly for each training data size. Yet as discussed in section 2.7 and 3.2.1, it is well accepted in literature that the performance of the

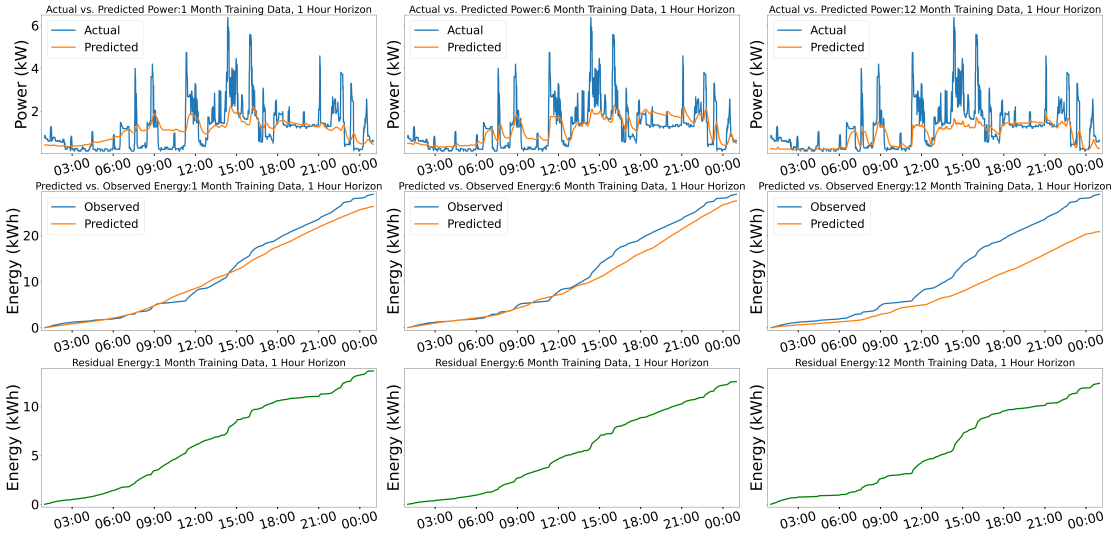


FIGURE 4.7: Comparison of predicted and observed power and energy for 1, 6 and 12-month training data sizes at a 1-hour prediction horizon

LSTM model is directly related to the data used to train it. This indicates that the required multi-step prediction was a challenge for the chosen LSTM architecture to perform. Despite the additional training data samples the neural network was unable to process additional training data to improve the model fit for increasing prediction horizon lengths.

4.2.2 Energy Analysis

To evaluate the energy prediction performance analyses of the predicted and residual energy values were determined for select models produced in Experiment B. Figure 4.7 shows the predicted and observed power profiles, energy for the 1, 6 and 12-month training data sizes at a 1-hour prediction horizon in Year 1.

Starting from the plots of predicted and observed power, it can be seen that the visual match between the predicted and observed profiles does not show overall improvement or degradation as the training data size increases from the 1 to 12-months.

Looking at the plots of the predicted and observed energy the 1 and 6-month models predict energy values of 26.8 kWh and 28.1 kWh which represent 8.8%

and 4.4% errors respectively. The 12-month model predicts 21.2 kWh which represents an error of 27.93%.

Lastly from the residual energy plot the 1, 6 and 12-month models each determined similar values of residual energy: 13.6 kWh, 12.5 kWh and 12.4 kWh respectively. These findings show that for an increasing prediction horizon the change in energy forecast accuracy varies similarly for each training data size. This is supported by findings in section 4.2.1 which showed similar behaviour in the change of MAE for different training data sizes. This indicates that for multi-step forecasts the relationship between training data size and prediction horizon length affects the power and energy prediction in similar ways. 12-month models are shown in table 4.4.

4.2.3 Power Peak Analysis

The observed profile and predicted profile peaks from the Experiment B models are compared to evaluate their peak prediction performance. A comparison of the observed power peaks vs. the predicted power profile on a 24-hour test data period for the 1, 6, and 12 month models over a 1-hour prediction horizon are shown in Figure 4.8. The MPE values for each model are similar, with values of 1.7 kW, 1.6 kW and 1.9 kW for the 1, 6 and 12-month models respectively. Inspection of each predicted profile also shows poor visual matches between the the observed and predicted peaks. This match does not noticeably improve between the predicted profiles of the 1 and 12-month models. A summary of the MPE for 1, 6 and 12-month models are shown in table 4.4. These findings show that the power peak forecast performance does not improve with additional training data over extended prediction horizon lengths.

Table 4.4: Summary of MPE values for 1, 6 and 12-month models for 1-hour horizon.

Training Data Size (months)	MPE (kW)
1	1.70
6	1.56
12	1.91

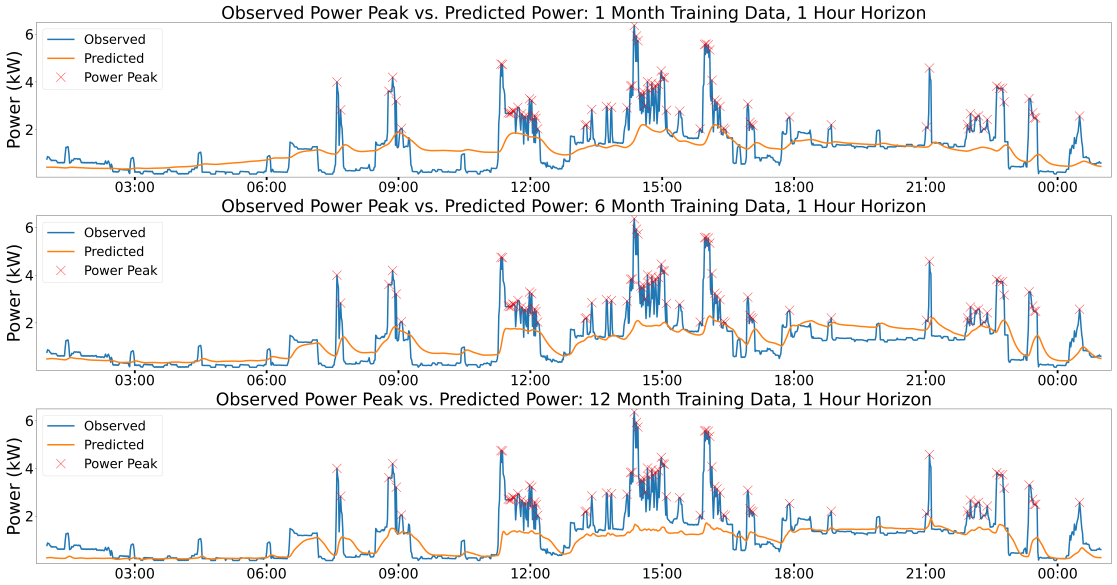


FIGURE 4.8: Comparison of observed peaks vs predicted power profile for the 1, 6 and 12-month prediction models.

4.2.4 Goodness of Fit

To evaluate the goodness of fit for each predicted profile an R^2 score was calculated for the models produced in Experiment B. As discussed in section 3.5.2 an R^2 score of one indicates an exact match between the predicted and observed profiles but a score of zero indicates that the predicted profile fits as well as a mean line of the observed values. Figure 4.9 shows a plot of the R^2 score against training data size for each of the models over a three-year duration.

Starting from the 1-minute prediction horizon of the Year 1 graph it can be observed that the R^2 score is at a maximum value of 0.88 for the 12-month training model. For the 1-minute horizon length the R^2 score is seen to decrease for each increase in training data size, reaching a minimum value of 0.54 at the 1-month model.

As the prediction horizon increases there is a general decrease in the R^2 score for models of each training data size. The rate of change for the R^2 score of model for each training data size vary similarly to each other.

At a horizon length of three hours it can be seen that the R^2 score is approximately at or below a value of zero for the majority of the training data size models. As the prediction horizon increases beyond three hours the R^2 scores for models of

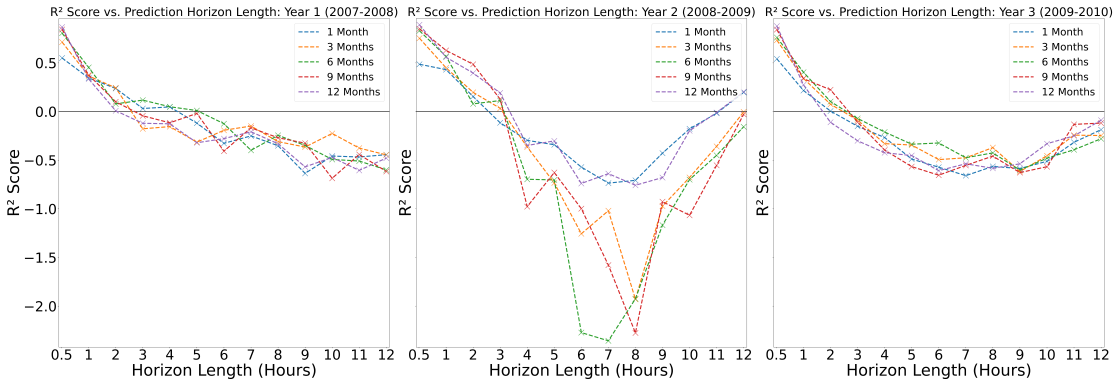


FIGURE 4.9: Comparison of R^2 scores for 1, 3, 6, 9 and 12-month training size models over three years.

each training data size continue to decrease below zero.

The trends in Year 2 are largely similar to those observed in Year 1. The largest R^2 scores are again observed at the 1-minute horizon length with minimum and maximum values of 0.49 and 0.89 at the 1 and 12-month models respectively. As the horizon length increases, similar variation in the change of R^2 scores for models of each training size are observed.

Also for prediction horizons larger than three hours the R^2 score for the majority models of each training data size are less than zero. Unlike the previous year, there is a sharp decrease in R^2 scores from the 5 to 8-hour horizons demonstrated by the 3, 6 and 9-month models. After the 10-hour horizon there is a slight increase in R^2 score across each model training size.

The trends and behaviours in MAE and horizon length for Year 3 are similar to those observed in Year 1 and Year 2: maximum R^2 scores at the 1-minute horizon, decreasing R^2 scores for increasing prediction horizon lengths and model scores falling at or below zero from the 3-hour horizon onwards.

The results from figure 4.9 show that from a horizon length of approximately three hours and longer the majority of prediction models have R^2 scores that are approximately equal to or less than zero. This indicates that these models no longer produce meaningful predictions, as their predicted profiles are comparable to or less accurate than using a mean of the observed values. Therefore, the effective prediction horizon range for this load profile is approximately three hours.

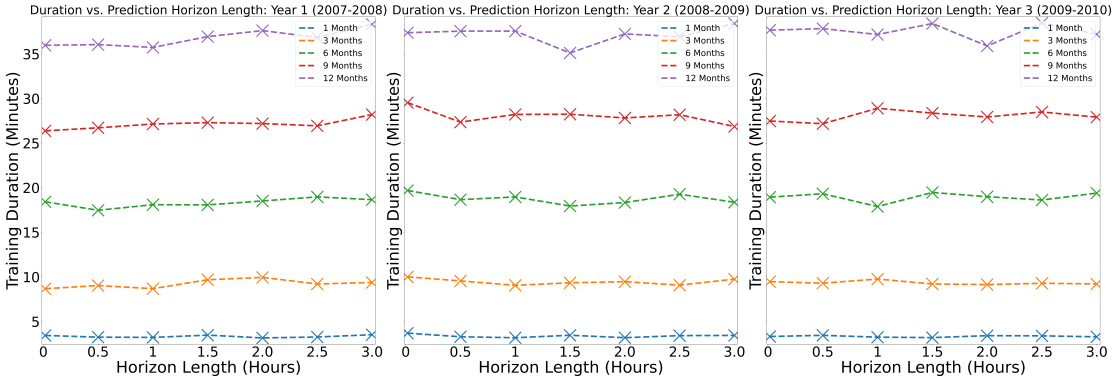


FIGURE 4.10: Comparison of training durations for each training data size up to a prediction horizon length of three hours

4.2.5 Model Training Duration

The model training duration was used to measure the compute intensity for the development of each prediction model. Figure 4.10 shows a plot of model training durations for each training data size up to the effective prediction horizon length of three hours as determined by the findings in section 4.2.4.

From figure 4.10 similar training duration behaviour can be observed in Year 1, 2 and 3. As the training data size increases the model training time also increases. An analysis of the mean training time for each training data size in figure 4.11 shows that for each year the minimum mean training times occur at the 1-month models and the maximum mean training times occur at the 12-month models.

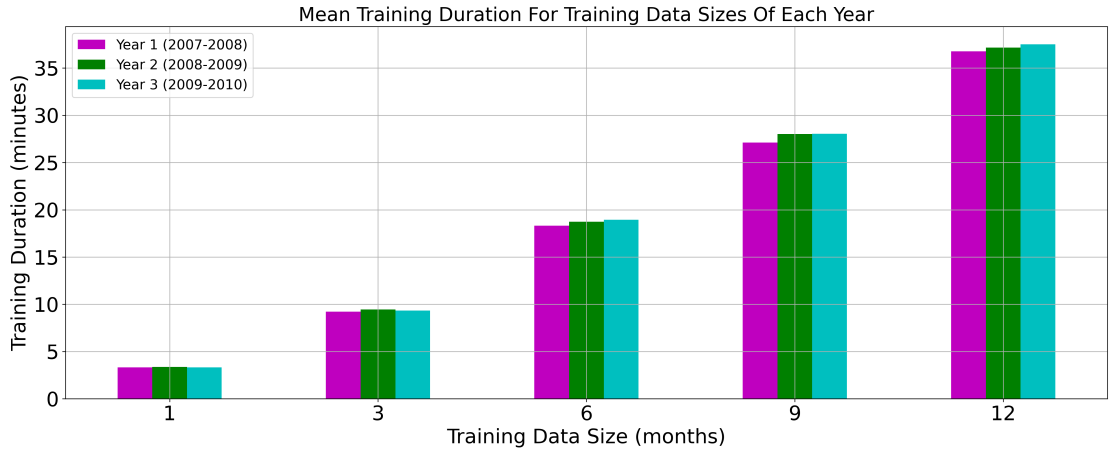


FIGURE 4.11: Comparison of mean model training time for each year of training data up to a horizon length of three hours.

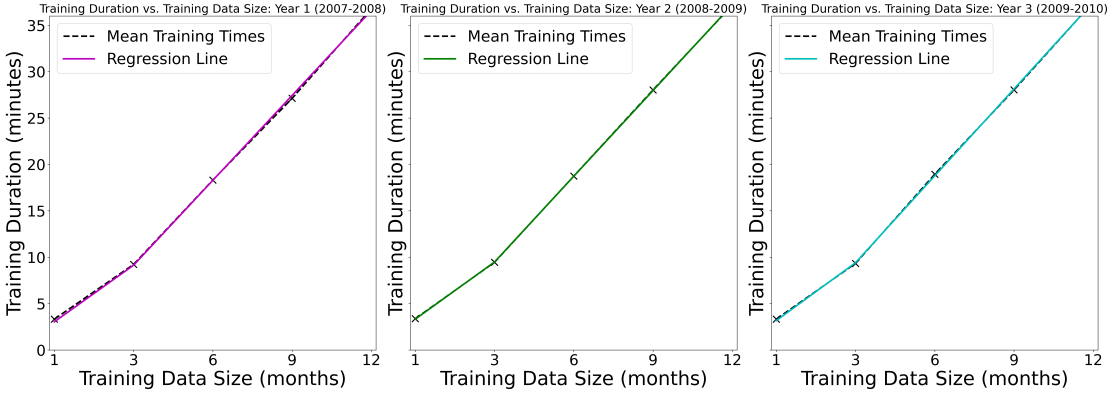


FIGURE 4.12: Regression analysis of mean model training time for each year of training data.

Furthermore, the regression analysis of the mean training times in figure 4.12 indicates an approximately linear relationship between the model training time and training data size. It can also be observed that as the prediction horizon length increases the training duration for each training data size does vary appreciably. These results show that there is a linear relationship between the model training time (i.e., compute intensity) and the training data size. This is consistent with the literature described in sections 3.4.2 and 3.2.1 which explains that additional training data samples require additional computations to update the neural network weights during training. These results also indicate that the model training duration is insensitive to changes in prediction horizon length.

4.2.6 Discussion

The change in MAE with increasing prediction horizon length observed in figure 4.5 indicates that the models across each training data year were unable to meaningfully predict the output. This can be seen by comparing the graphs of MAE and R^2 score vs. prediction horizon length as shown in figure 4.13. The goodness of fit analysis shows that for a prediction horizon of 3 hours the R^2 scores for the majority of prediction models are at or below a value of zero. This means that the predictive models with horizons longer than 3 hours are less effective than taking the mean of the observed values. Therefore the extent of the effective prediction horizon length for the models trained on this household load data is 3 hours.

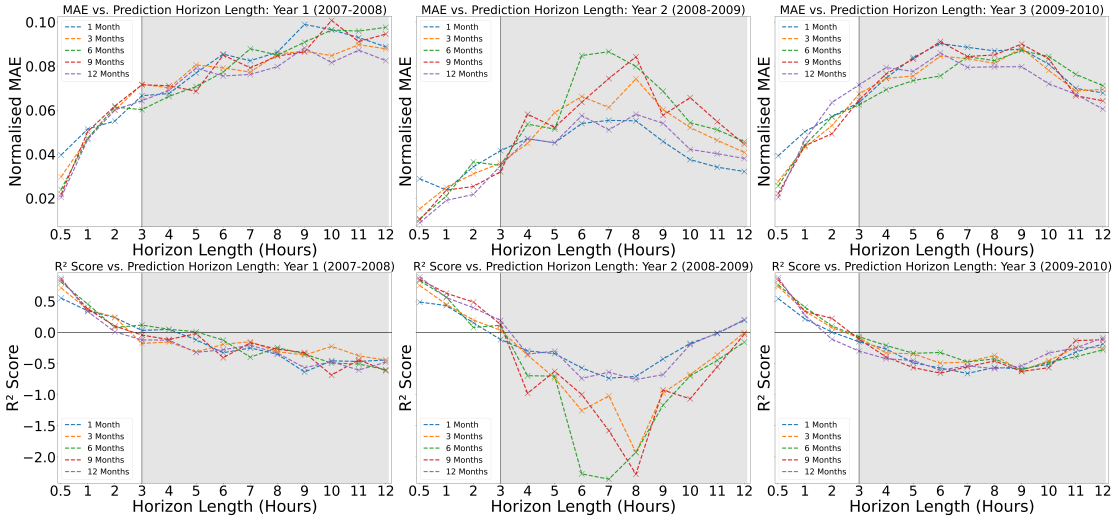


FIGURE 4.13: Illustration of the MAE and R^2 scores within the 3 hour prediction horizon length.

The MAE for some predictive models using prediction horizons greater than 3 hours are similar to models within the effective prediction length. A comparison of the observed and predicted power for 2 hour and 10 hour horizon length models with similar MAEs (0.034 and 0.038 respectively) is shown in figure 4.14. Despite having MAEs that are similar, the goodness of fit analysis shows the 10 hour prediction model performs worse than the 2 hour model. This can be observed in figure 4.14 where the 2 hour model predictions more closely follow changes in the observed power and power peaks. By comparing both the MAE and R^2 scores it can be seen that apparent improvements in the MAE beyond the 3 hour horizon are not indicative of better model performance but rather predictions incidentally resulting in smaller MAEs. This could be indicative of the predictive limitations of the chosen LSTM architecture and would require changes to the architecture and hyperparameter settings to improve the effective prediction horizon length.

In terms of load forecasting applications, a GEB system with an effective prediction horizon of 3 hours is suitable for short-term load forecasting. This includes residential DER management which requires coordination between DER generation, minutely to hourly changes in user consumption, and RTP tariffs which changes hourly [191, 192].

Analyses of the power, energy and peak prediction errors for the Experiment B models demonstrated that the addition of training data alone is not sufficient to

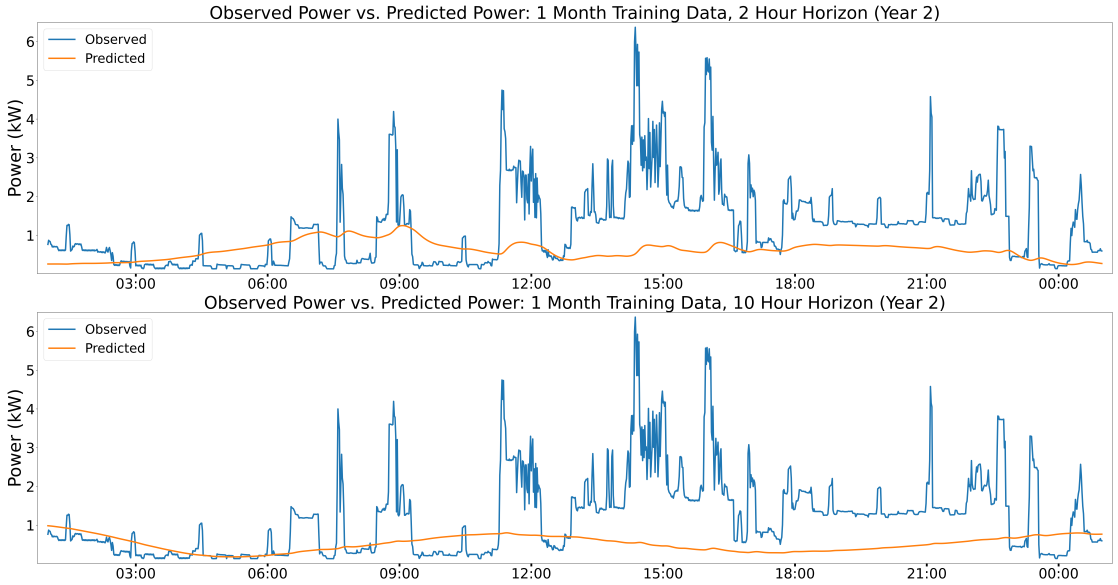


FIGURE 4.14: Comparison of observed and predicted power between 2-hour and 10-hour horizon length models.

improve the model performance for extended prediction horizons. Yet, it is well accepted in the literature that the training data directly impacts the model performance, and that LSTM neural networks are capable of performing accurate forecasts with extended prediction horizons [176, 177]. This further suggests that the LSTM architecture chosen for this experiment does not process the training data in such a way that additional training data reflects improvements in model performance for multiple time steps. This means that, for a GEB, the specific model architecture is a contributing factor for effective multiple time step load forecasting in addition to the amount of training data.

The results of the model training duration analysis show that the computational intensity increases with training data sizes, but does not vary appreciably as the prediction horizon increases. This means that for this household and LSTM architecture it is more resource efficient to predict longer horizons using smaller amounts of training data.

4.2.7 Summary of Findings

The findings of Experiment B show that as the prediction horizon length increases the goodness of fit between the predicted and observed profiles for each training

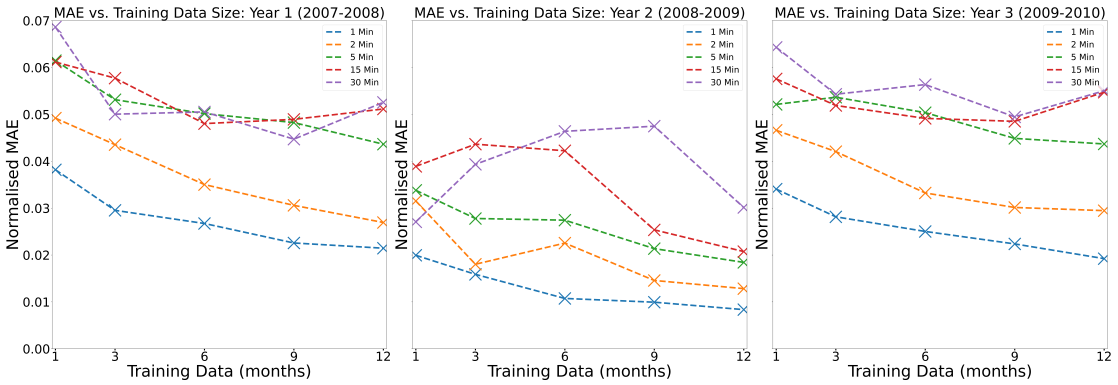


FIGURE 4.15: Comparison of model prediction errors and training data size for various data resolutions.

data size diminishes at similar rates. This relationship enabled the identification of an effective prediction horizon length of three hours for the load profile under investigation. It was found that the chosen LSTM architecture was not able to reflect improvements in model performance for additional training data. The relationship between the training data size and the change model prediction errors for extended horizon lengths is unclear.

It was shown that the computational intensity increases with training data sizes, but does not vary appreciably as the prediction horizon extends.

4.3 Experiment C: Training Data Resolution and Training Data Size

Experiment C investigates the relationship between the change in model performance for different data resolutions as the training data amount increases. This section details the experimental results and provides analyses of the model prediction error and goodness of fit, as well as the residual energy and power peak prediction performance.

A comparison of model prediction errors against the training data size for five different resolutions is shown in figure 4.15. Each plot in the figure represents results obtained for training data originating from different years.

Starting from the Year 1 plot it can be observed that the MAE of models using 1 and 2-minute resolutions are the lowest for each training data size. The maximum

and minimum MAEs for the 1-minute resolution are 0.034 and 0.019 respectively and for the 2-minute resolution are 0.047 and 0.029 respectively. This indicates that for this year the 1 and 2-minute resolution models have the highest prediction accuracy compared to the lower resolution models i.e., 5, 15 and 30-minutes. It can also be observed that the MAE decreases for each additional training data size. The additional 11 months of training data between the 1-month and 12-month models produce a 45% and 38% reduction in MAE for the 1 and 2-minute resolutions respectively.

Furthermore, it can be observed that the 2-minute models trained on larger data sizes begin to produce MAE values comparable to those of 1-minute resolution models. This is seen at the 6-month data size model with an MAE of 0.035 which is similar to the 1-minute resolution model MAE of 0.038 which occurs at 1-month of data. This shows that lower resolution models can achieve similar performance to higher resolution models given additional training data.

For the 5, 15 and 30-minute resolution models it is observed that their respective MAE values are in close proximity to each other across each training data size compared to the 1 and 2-minute models. Similar MAE values for models of different resolutions can be observed between the 5 and 15-minute models for the 1-month data size with MAEs of 0.052 and 0.058 respectively, and between the 5 and 30-minute models for the 6-month data size with MAEs of 0.045 and 0.049 respectively. This suggests that models for these three resolutions produce similar prediction performance.

Like the 1 and 2-minute resolution models the 5-minute model MAEs decrease with every training size increase, with a maximum MAE of 0.052 and a minimum MAE of 0.044. Yet, the 15 and 30-minute models do not follow this trend and demonstrate increases in error for some of the larger training data sizes. For the 15 and 30-minute resolutions increases in MAE can be observed from the 6 to 12-month data sizes and the 9 to 12-month data sizes respectively. This indicates that for lower resolutions additional training data does not necessarily result in improved prediction performance.

In Year 2 the 1 and 2-minute resolution models demonstrate behaviour similar to

the previous year: having the lowest MAE for models across each training data size with maximum and minimum MAEs for the 1-minute resolution of 0.020 and 0.008 respectively and for the 2-minute resolution 0.020 and 0.008 respectively. The MAE also decreases for increases in training data size with 60% and 59% reductions in MAE between the 1 and 12-month training data sizes for the 1 and 2-minute resolutions respectively.

Compared to Year 1 the prediction errors for the 5, 15 and 30-minute resolution models are more dissimilar to each other than in Year 1 with no overlaps in prediction error observed. This indicates a change in their relative predictive performances.

The 5-minute resolution models demonstrate similar behaviour to those in Year 1 with the exception of producing lower MAEs compared to the 15 and 30-minute models. This indicates improved performance of the 5-minute models and suggests that the Year 2 data is more effective for slightly lower resolution models than Year 1.

The 15 and 30-minute resolution models initially exhibit high MAE values between from the 1 to 6-month training data sizes and then begin to reduce, indicating improving performance. Models produced from the lower resolution remain the least accurate, but unlike Year 1 the MAE for the lower resolution models improves for larger training data sizes.

In Year 3 the behaviour and patterns for models of each resolution and training data size are similar to those observed in Year 1.

4.3.1 Model Prediction Error Analysis

The results from figure 4.15 indicate three key relationships between the model prediction accuracy, training data size and data resolution.

Firstly, the model performance decreases as the data resolution reduces. An analysis of the mean difference in MAE relative to the 1-minute resolution models is shown in figure 4.16. It can be seen that the models trained on the 2-minute resolution data produce relatively small mean difference values across each year.

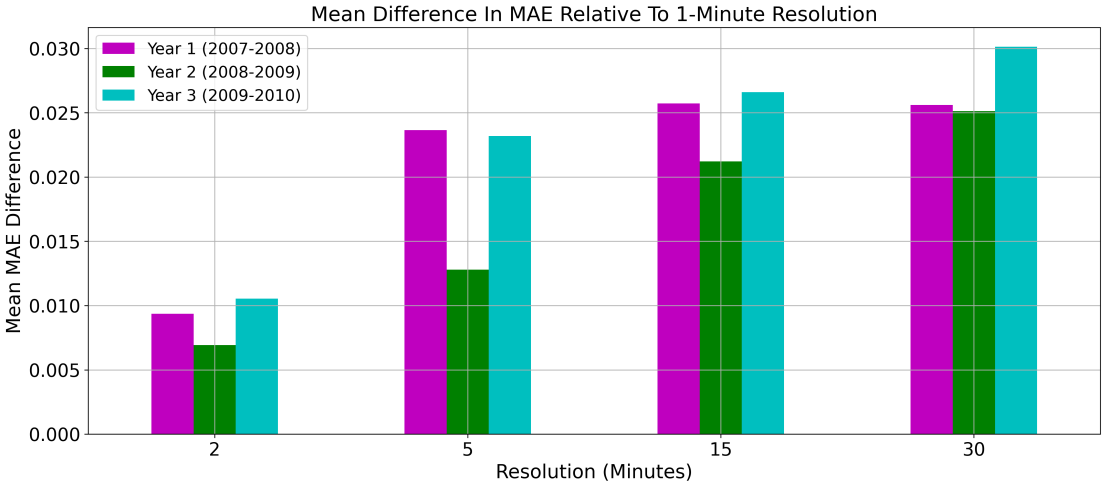


FIGURE 4.16: Mean Difference In MAE For Lower Resolution Models Relative to 1-Minute Resolution Models.

As the model resolution decreases the mean difference increases, reaching a peak value at a resolution of 30 minutes.

Secondly, models trained on low (i.e, 5, 15 and 30-minute) resolution data can achieve prediction performance similar to that of high (i.e., 1 and 2-minute) resolution models when trained on larger sets of data. For example in Year 1, the 2-minute resolution model with six months of training data has an MAE of 0.035 which is similar to the MAE of 0.038 for the 1-minute model trained on one month of data. Similarly, the 15-minute resolution model with six months of training data has an MAE of 0.048 which is less than the MAE of 0.050 for the 5-minute model trained on the same amount of data. As the training data increases the MAE for all model resolutions generally decreases. As a consequence, the MAE for low resolution models at larger training data sizes can decrease to values less than or comparable to the MAEs of the high resolution models achieved using the same or smaller training data sizes.

From a study of the impact of training data size on LSTM performance by Boulmaiz et al and the results of Experiment A in section 4.1 it has been found that additional training data generally improves the performance of prediction models [4]. A similar behaviour is observed in figure 4.15 as the model performance generally increases with larger training data size for each resolution. The introduction of additional predictor (i.e., feature) samples enable the LSTM network to

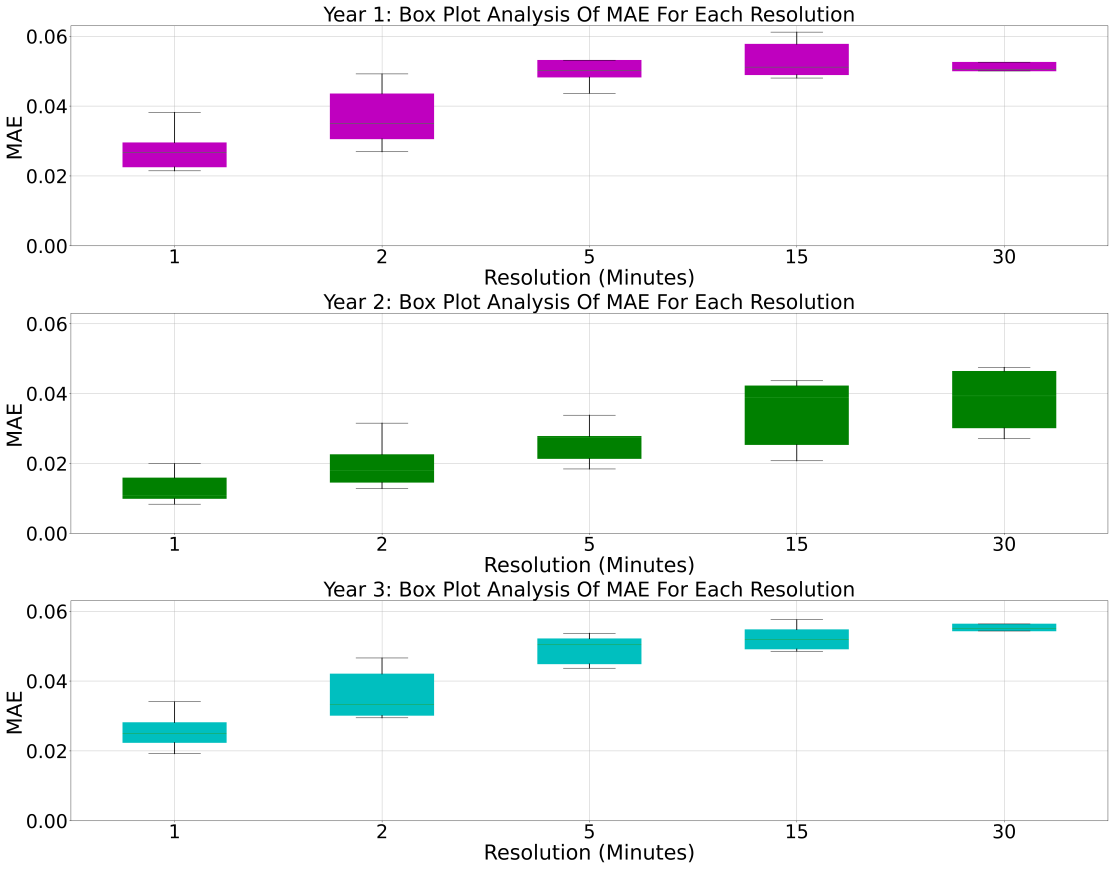


FIGURE 4.17: Box plot analysis of MAE distributions for models of each resolution.

more effectively update the neural network weights, which produces more accurate models.

Lastly, models trained on lower resolution (i.e., 5, 15 and 30-minute) data demonstrate similar prediction performance to each other when compared to the higher resolution (i.e., 1 and 2-minute) models. A box plot analysis comparing the MAEs for each resolution size is shown in figure 4.17. It can be seen from figure 4.17 that in Year 1 and 3 the low resolution models share similar median values that are larger than that of the high resolution models. In Year 3 the 5-minute resolution shows a distribution more closely aligned to that of the high resolution models, but for the 15 and 30-minute models their median and interquartile ranges are still closely matching.

As discussed in section 3.4.3 using a higher resolution captures the changes in power usage more accurately than using a lower resolution. The box plots of the model MAEs in figure 4.17 show that the profile patterns of the 1 and 2-minute

models are statistically similar to each other and distinct from the patterns of the 5, 15 and 30-minute resolution models.

4.3.2 Energy Analysis

To evaluate the energy prediction performance analyses of the predicted and residual energy values were determined for select models produced in Experiment C. Figure 4.18 shows the predicted and observed power profiles, energy and residual energy for the 1, 5 and 30-minute resolution models trained on 360 days of data in Year 1.

From the plots of predicted vs actual power it can be observed that the visual match between the observed and predicted power diminishes as the resolution decreases from 1 to 30-minutes. The visual match for the 1-minute resolution model shows very small differences between the predicted and observed plots, demonstrating good model performance. As the resolution decreases the predicted profiles start to fit the observed data points more poorly. This indicates a decline in the predictive model performance for decreasing data resolution.

The value of observed energy is 28.9 kWh. The 1-minute resolution model predicts an energy value of 25.1 kWh which is a 13.1% error from the observed energy. As the data resolution decreases the predicted energy error increases, reaching a maximum error of 43.0% at a resolution of 30-minutes.

Lastly, from the plot of residual energy the amount of incorrectly predicted energy is at a minimum value of 5.7 kWh at the 1-minute resolution. A similar pattern is observed where the residual energy increases as the data resolution decreases. A maximum residual energy value of 13.2 kWh occurs at the 30-minute resolution model.

The results from the energy analysis indicate that the model energy forecasting ability degrades as the resolution decreases. In table 4.5 a summary of the residual energy for each of the models including their respective MAE values from figure 4.15 are shown.

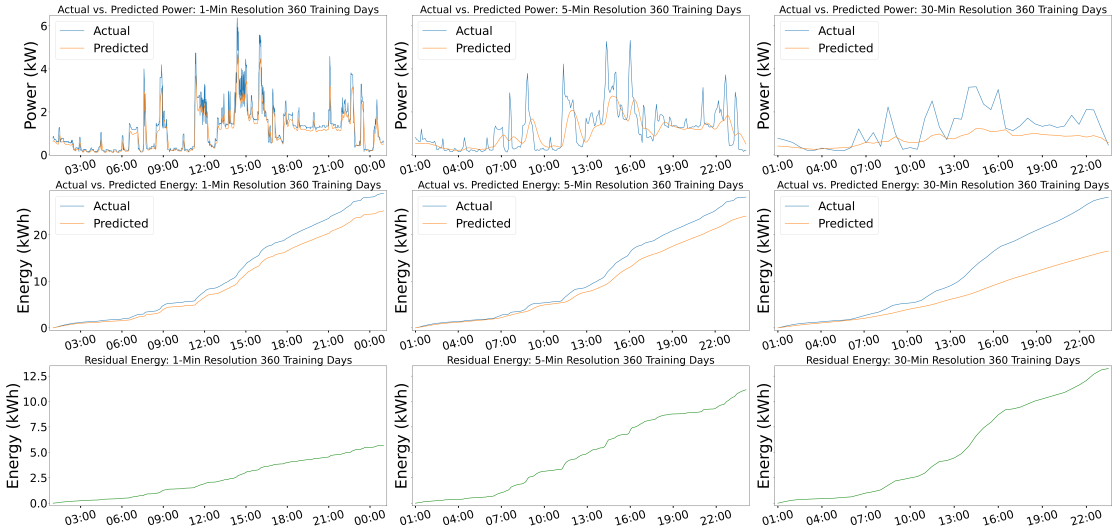


FIGURE 4.18: Comparison of predicted and observed power and energy for 1, 5 and 30-minute resolution models.

4.3.3 Power Peak Analysis

To evaluate the power peak, an MPE score was calculated for the models produced in Experiment C. Figure 4.18 shows the observed power peaks and predicted power profiles for the 1, 5 and 30-minute resolution models trained on 360 days of data in Year 1.

For a 1-minute resolution the MPE is at a minimum of 0.48 kW. Visual inspection shows that at this resolution the predicted profile closely matches the observed peaks.

As the resolution decreases the MPE increases, with a value of 1.1 kW observed at the 5-minute resolution and a maximum MPE of 1.2 kW at the 30-minute resolution. Visual inspection of the observed and predicted profiles shows increasingly poor matches between the observed and predicted profiles for the 5 and 30-minute resolutions.

TABLE 4.5: Summary of power and energy predictions for select models.

Resolution (minutes)	Observed Energy (kWh)	Residual Energy (kWh)	MAE	% Residual Energy
1	28.94	5.7	0.021	19.7
5	28.12	11.1	0.044	38.5
30	28.11	13.2	0.053	45.8

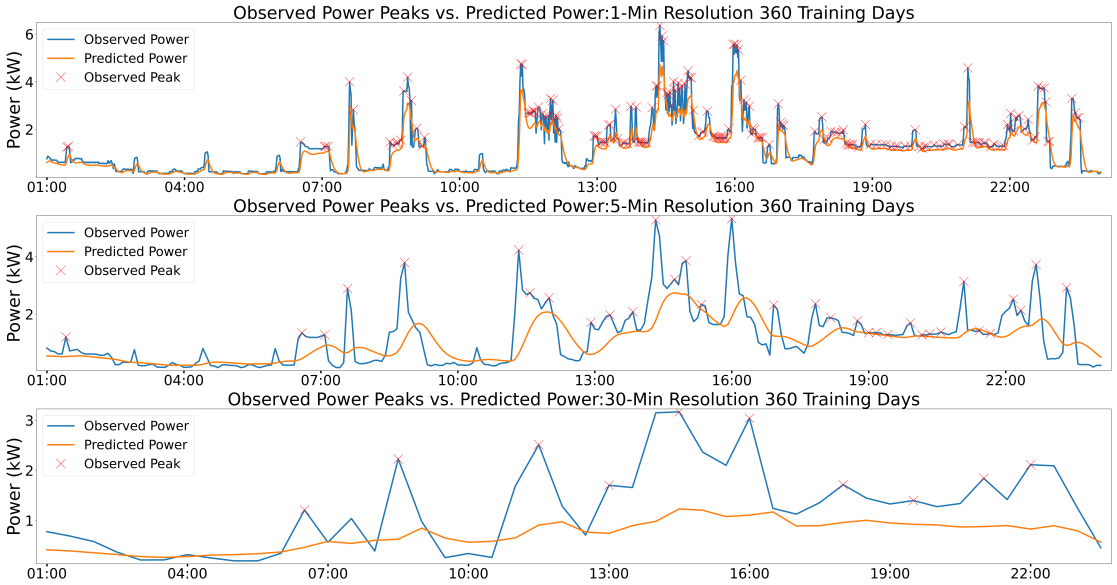


FIGURE 4.19: Comparison of the observed power peaks and predicted power for the 1, 5 and 30-minute resolution models

These findings show that the ability for the models to predict large power peaks diminishes as the resolution decreases. In table 4.6 a summary of the MPE for each model including their respective MAE values from figure 4.15 are shown.

4.3.4 Goodness of Fit

To evaluate the prediction profile goodness of fit, an R^2 score was calculated for each model produced in Experiment C. As discussed in section 3.5.2 an R^2 score of one indicates an exact match between the predicted and observed profiles but a score of zero indicates that the predicted profile fits as well as a mean line of the observed values. Figure 4.20 shows a comparison of R^2 against training data size for each of the models over a three year duration.

Starting from Year 1 the R^2 scores show that the 1 and 2-minute resolution models

TABLE 4.6: Summary of power peak predictions for select models.

Resolution (minutes)	MPE (kW)	No. of Peaks	MAE
1	0.48	185	0.021
5	1.05	34	0.044
30	1.22	10	0.053

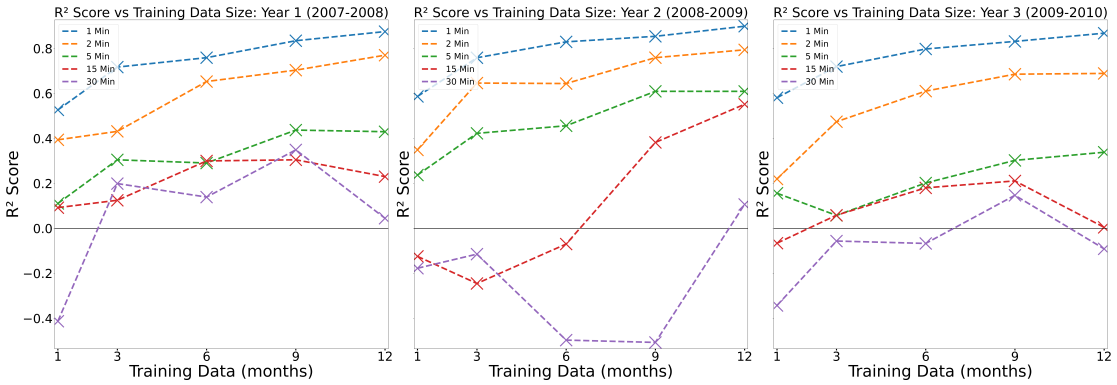


FIGURE 4.20: Comparison of R^2 scores for 1, 5, 15, and 30-minute resolution models over 3 years.

follow similar trends.

The scores for the 1 and 2-minute resolution models are the highest for each training data size. The maximum and minimum MAEs for the 1-minute resolution are 0.53 and 0.88 respectively and for the 2-minute resolution are 0.39 and 0.77 respectively. This shows that the predicted profiles of the 1 and 2-minute models provide the closest fit to the observed profile compared to the other resolution sizes. It can also be observed that the R^2 score improves for larger training data sizes with a 66.1% improvement from the 1 to 12-month training size for the 1-minute resolution and a 95.5% for the 2-minute resolution.

For the lower data resolutions (i.e., 5, 15 and 30 minutes) it can be seen that there are larger reductions in the R^2 score compared to the higher resolutions (i.e., 1 and 2 minutes). This indicates that the predicted profile fit degrades for models at lower data resolutions.

The minimum and maximum R^2 scores of the 5-minute resolution models are 0.11 and 0.43 respectively. There is 79.2% decrease between the maximum R^2 scores of the 2 and 5-minute models. This difference is over 5 times greater than the 13.6% reduction in maximum R^2 scores between the 1 and 2-minute models.

At 30-minutes the model performance is at its poorest. The model R^2 scores begin to approach values at or below zero, with values of -0.41 and 0.04 at the 1 and 12-month training sizes respectively. This indicates severe reductions in prediction performance as the predicted profiles from these models are about as accurate as taking the mean of the observed profile values.

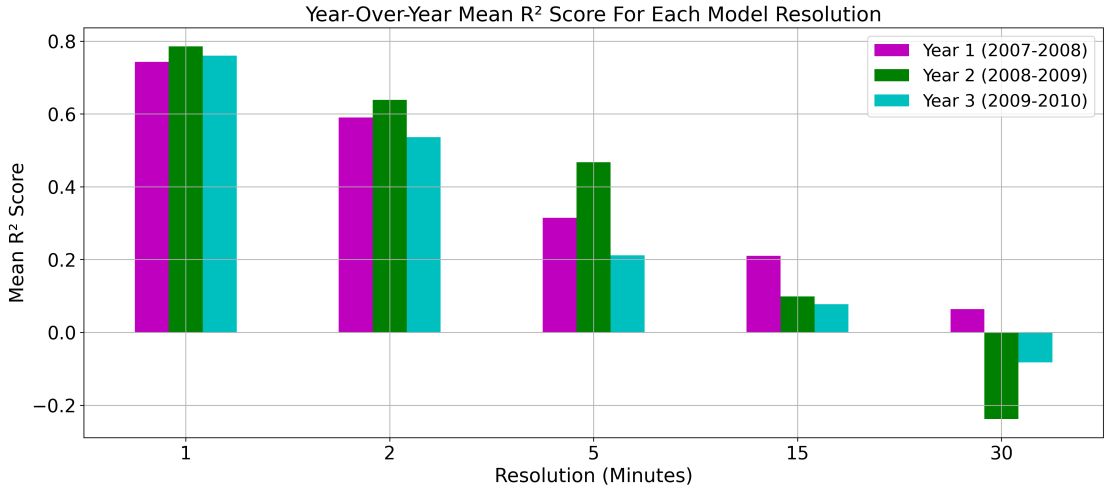


FIGURE 4.21: Mean R^2 scores of each year for 1, 5, 15, and 30-minute resolution models.

From the Year 2 scores it can be seen that the 1 and 2-minute resolution models exhibit similar behaviour to the previous year, with the highest and second highest R^2 scores respectively, which also increase with increases in training data size.

The 3-minute resolution models show improved performance over the models of the previous year with 54.2% and 41.9% increases in the minimum and maximum R^2 scores respectively.

The 15-minute resolution models of Year 2 show a reduction in performance compared to the previous year, with over 60% of the models having R^2 scores that are below zero. Similarly the 30-minute resolution models demonstrate a sharp decline in performance with 80% of them attaining R^2 scores below zero.

In Year 3 the trends and behaviour for models of each data resolution and training data size are largely the same as those observed in Year 1.

To quantify the performance of each model resolution in figure 4.20 a mean of the R^2 scores was calculated for each year of training data. The results are shown in figure 4.21. It can be seen that the R^2 score mean for each year decreases as the model resolution decreases; with a maximum mean score of 0.79 at a 1-minute resolution and a minimum mean score of -0.24 at 30-minutes. These results corroborate the previous observations of figure 4.20 which indicated that a lower model resolution reduced the model performance.

4.3.5 Discussion

Analysis of the prediction errors in figure 4.3 showed that the model performance decreases as the data resolution decreases. For GEB load forecasting this means that there is a trade-off between forecasting quality and measurement device costs. Using lower resolution devices generally costs less than higher resolution devices but will result in less accurate prediction models.

Yet, an additional finding from these results show that the reduced performance of lower-resolution models can be improved to some extent with additional training data. This means the trade-off between prediction performance and device cost can be mitigated using larger amounts of training data. Lower resolution devices can produce models that achieve performance comparable to models trained on higher resolution data. Furthermore by comparing the distribution of model errors for each data resolution the extent to which a lower resolution model can approximate higher resolution performance can be estimated. For instance, in Year 1 of figure 4.1 the performance of the 2-minute resolution models becomes comparable to the 1-minute resolution after 6 months of additional training data. This means that the efficacy of a GEB load forecasting system that uses cheaper measuring devices can improve appreciably over time.

The results in figure 4.1 show that between the 30 and 1-minute resolution models there is a change in MAE from 0.053 to 0.021 which is a 60.4% reduction. From the energy and power peak analysis, this corresponds to a change in residual energy from 45.8 kWh to 19.7 kWh which is a 57% reduction, and a change in MPE from 1.2 kW to 0.48 kW which is a 154% decrease. This indicates that reductions in power prediction error produce appreciable improvements in the energy and power peak prediction performance. Conversely, as the model prediction performance degrades at lower resolutions the amount of erroneously predicted energy increases. This means that the trade-off between power prediction accuracy and device cost also significantly impacts the ability to participate in energy-sensitive applications such as TOU tariffs and energy markets.

The findings from figure 4.20 and figure 4.21 indicate that the predicted models fit

the observed data more poorly as the model resolution decreases. For a 30-minute resolution it can be observed that the R^2 score achieves negative values and it can be inferred that resolutions less than 30 minutes would produce even lower scores. This relationship between the model resolution and the R^2 score is useful as an assessment tool to estimate the extent to which the resolution of a GEB load profile can be reduced while still producing accurate prediction models. As explained in section 3.5.2 an R^2 score less than or equal to zero means that the predicted model fit is comparable to or worse than a mean line of the observed values. This indicates that the model does not meaningfully predict the target variable and is not viable. Therefore, the zero-line for the plot of R^2 score vs. training data size can be used as a threshold to estimate the minimum viable measurement resolution and thereby determine the minimum appropriate measuring device resolution.

4.3.6 Summary of Findings

The Experiment C findings show the model performance decreases for models trained on lower data resolutions. In addition it is found that lower resolution models can achieve similar performance to higher resolution models when trained on larger datasets. The relationship between data resolution and training data size enabled the estimation of the extent to which lower resolution models can approximate higher resolution model performance. Furthermore, the goodness of fit was found to decrease for predicted profiles of models trained on lower resolution data. This finding showed that the load profile under investigation has a minimum viable data resolution of 30 minutes. It is also shown that reductions in data resolution produce appreciable degradations in the energy and power peak prediction ability of each model.

4.4 Chapter Summary

In this chapter the results of the three experiments used to investigate the relationships between several design parameters and their impact on model fidelity

are analysed and discussed.

Experiment A examines the extent to which increasing amounts of training data can improve model prediction accuracy and the computational cost required. The relationships between training data size and model performance, as well as training data size and compute intensity are shown to be approximated by an exponential decay and linear functions respectively. This finding enabled the identification of a resource-efficient upper training data size limit of 450 days.

Experiment B examines the extent to which an increase in training data amount can improve the model accuracy for increasingly large prediction horizons and the computational cost required. The relationships between the training data size and prediction horizon length showed similar reductions in goodness of fit between the predicted and observed profiles across each training data size. This finding enabled the identification of an effective prediction horizon range of 3 hours.

Experiment C examines the relationship between the change in model performance for different data resolutions as the training data amount increases, and the change in computational resources required for training models on different data resolutions. It is shown that the model performance decreases as the training data resolution decreases, but lower resolution models can approximate the model performance of higher resolution models by using larger training data sizes. These relationships enable an estimation of the extent to which lower resolution models can improve their model performance for a given set of training data.

CHAPTER 5

Conclusion And Future Work

This chapter presents a summary of results and highlights key findings from experimental work in section 5.2. Discussions of improvements and future extensions of this work are presented in section 5.3. Finally the concluding remarks for this dissertation are presented in section 5.4.

5.1 Introduction

The objective of this research work was the investigation of the following question: *To what extent can the relationship between machine-learning design parameters and their predictive model fidelity inform model design choices for practical load profile applications?*

The following objectives were formulated to answer this question:

- Investigate the relationships between training data size, prediction model error and compute intensity.
- Investigate the relationships between prediction horizon length, training data size, model prediction error and compute intensity.
- Investigate the relationships between data resolution, training data size and prediction model error.

This chapter details the results from this work and links the findings to the research question and set objectives.

5.2 Findings

The key findings from this work are summarised as follows:

- The training data size parameter is shown to have an exponentially decaying relationship with the prediction model performance and a linearly increasing relationship with the computational intensity of model training from equations 4.1 and 4.2 respectively. These relationships can be used to identify the extent to which the training data size can be increased before improvements in model performance become disproportionately small compared to the computational resource cost required to train the model. For the load profile used in this investigation this was found to be a training data size of 450 days. *(Chapter 4, section 4.1.1)*

- The statistical dispersion of measurements within the load profile data are shown to reduce the model prediction accuracy. *(Chapter 4, section 4.1.4)*

- The R^2 score of the predicted load profiles is shown to decrease as the prediction horizon parameter increases. This relationship can be used to identify the extent to which the horizon can be extended while still producing viable predictions. For the load profile and LSTM architecture used in this investigation the effective prediction horizon length was found to be 3 hours. *(Chapter 4, section 4.2.4)*

- Prediction performance is shown to degrade as the data resolution hyperparamater for each model decreases, but lower resolution models can approximate the performance of models trained on higher resolution data by increasing the training data size. The relationship between the data resolution and model prediction error can be used to estimate the extent to which a lower resolution model can achieve the performance of a higher resolution model. *(Chapter 4, section 4.3.1)*

- The R^2 score of the predicted load profiles is shown to decrease as the data resolution hyperparamater decreases. This relationship can be used to identify the minimum viable data resolution. For this load profile the minimum data resolution is found to be approximately 30 minutes. *(Chapter 4, section 4.3.4)*

- Improvements in the power prediction performance are shown to produce appreciable changes in the models' energy forecast accuracy. *(Chapter 4, section 4.1.3 and 4.3.3)*

5.3 Future Work

The LSTM architecture used for this work was sufficient to produce viable prediction models for experimentation. However, future implementations of this architecture should be improved to deliver higher performance output:

- Include additional training features such as temperature, weather conditions and home occupancy to improve model performance.

- Investigate alternative LSTM architectures and hyperparameter settings suited to extended prediction horizon forecasting. Longer effective prediction horizons enable more applications for GEBs such as automated energy market management.

- Consider alternative loss functions for model training.

The experiments performed in this investigation have been shown to identify several relationships between model design parameters and the model fidelity. However some improvements for this experimentation can be considered:

- Perform frequency-domain analyses of the load profile data to determine the effects of power usage frequency on model performance.

- Examine and compare design parameter relationships for load profiles of similar homes from different geographical locations.
- Consider real TOU tariff rates to improve evaluation of the energy prediction performance.
- Use load profile data spanning a longer time period.
- Include an additional experiment to investigate changes in model performance for different rates of model re-training frequency.

Exploration of this work could be considered for the following future applications:

- Classify load profiles based on residential features and examine to what extent the design parameter relationships generalise across different residence classes.
- Investigate the integration of the load forecasting models as part of a residential energy trading management system.

5.4 Conclusion

In concluding this work, the three research objectives and how they address the research question are discussed.

The first objective was to investigate the relationships between training data amount, prediction model error and compute intensity. To achieve this, Experiment A was conducted to develop a series of power prediction models each developed using an increasingly large amount of training data. The model prediction error and compute intensity (in terms of model training time) were measured. It was shown that for the load profile under investigation the training data size and model prediction error relationship can be modelled by an exponential decay function, and the training data size and compute intensity relationship can be modelled by a linear function.

The second objective was to investigate the relationships between prediction horizon length distance, training data amount, model prediction error and compute intensity. To achieve this, Experiment B was conducted to develop a series of power prediction models developed on increasingly large sets of training data, using increasingly long prediction horizon lengths. The model prediction error and compute intensity (in terms of model training time) were measured. It was found that for each training data size, as the prediction horizon length increased the goodness of fit between the predicted and observed profiles diminished at similar rates. The relationship between the training data size and model prediction errors for extended prediction horizon lengths was unclear. Future work can improve the LSTM architecture for more effective extended prediction horizon forecasts.

The third objective was to investigate the relationships between data resolution, training data size and prediction model error. To achieve this, Experiment C was conducted to develop a series of power prediction models developed on increasingly large sets of training data of different resolutions. The model prediction error and compute intensity (in terms of model training time) were measured. It was shown that the model prediction error increases for models trained on decreasing data resolutions. Furthermore it was found that lower resolution models can achieve model prediction errors comparable to higher resolution models by using larger training data sizes.

These three objectives were derived from the research question: *To what extent can the relationship between machine-learning design parameters and their predictive model fidelity inform model design choices for practical load profile applications?* Based on the results from this experimental investigation it has been affirmed that the relationships between the model prediction error and training data size, prediction horizon length and data resolution parameters can be used to inform model design choices including: maximum resource-effective training data size, effective prediction horizon length, minimum viable data resolution and the extent of low-resolution model improvement for additional training data.

Therefore this work represents a contribution towards the development of a ML model design tool for more efficient design parameter tuning in practical load

forecasting environments. This shows potential to enable more time and resource efficient implementations of load forecasting systems for GEB applications.

References

[1] IEA, "Electricity market report 2023," 2023. Accessed on March 18, 2024.

[2] IEA, "Electricity information: Overview," 2021. Accessed on March 18, 2024.

[3] S. Gyamfi, S. Krumdieck, and T. Urmee, "Residential peak electricity demand response—highlights of some behavioural issues," *Renewable and Sustainable Energy Reviews*, vol. 25, pp. 71–77, 9 2013.

[4] T. Boulmaiz, M. Guermoui, and H. Boutaghane, "Impact of training data size on the lstm performances for rainfall–runoff modeling," *Modeling Earth Systems and Environment*, vol. 6, pp. 2153–2164, 12 2020.

[5] G. Hayder, M. I. Solihin, and M. R. Najwa, "Multi-step-ahead prediction of river flow using narx neural networks and deep learning lstm," *H2Open Journal*, vol. 5, pp. 43–60, 3 2022.

[6] L. Zhou, C. Zhao, N. Liu, X. Yao, and Z. Cheng, "Improved lstm-based deep learning model for covid-19 prediction using optimized approach," *Engineering Applications of Artificial Intelligence*, vol. 122, p. 106157, 6 2023.

[7] X. Wang, H. Wang, B. Bhandari, and L. Cheng, "Ai-empowered methods for smart energy consumption: A review of load forecasting, anomaly detection and demand response," *International Journal of Precision Engineering and Manufacturing-Green Technology 2023*, pp. 1–31, 9 2023.

[8] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches," *Energies 2018, Vol. 11, Page 1636*, vol. 11, p. 1636, 6 2018.

[9] M. G. Abdolrasol, S. M. S. Hussain, T. S. Ustun, M. R. Sarker, M. A. Hannan, R. Mohamed, J. A. Ali, S. Mekhilef, and A. Milad, "Artificial neural networks based optimization techniques: A review," *Electronics 2021, Vol. 10, Page 2689*, vol. 10, p. 2689, 11 2021.

[10] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh, and B. B. Chaudhuri, "Diffgrad: An optimization method for convolutional neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, pp. 4500–4511, 11 2020.

[11] Y. Yang, J. Liu, Y. Lin, and Q. Li, "The impact of urbanization on china's residential energy consumption," *Structural Change and Economic Dynamics*, vol. 49, pp. 170–182, 6 2019.

[12] S. Franco, V. R. Mandla, and K. R. M. Rao, "Urbanization, energy consumption and emissions in the indian context a review," *Renewable and Sustainable Energy Reviews*, vol. 71, pp. 898–907, 5 2017.

[13] T. J. Nechyba and R. P. Walsh, "Urban sprawl," *Journal of Economic Perspectives*, vol. 18, pp. 177–200, 9 2004.

[14] P. Zhao and M. Zhang, "The impact of urbanisation on energy consumption: A 30-year review in china," *Urban Climate*, vol. 24, pp. 940–953, 6 2018.

[15] P. Bairoch and G. Goertz, "Factors of urbanisation in the nineteenth century developed countries," *Urban Studies*, vol. 23, pp. 285–305, 1986.

[16] D. Castells-Quintana and H. Wenban-Smith, "Population dynamics, urbanisation without growth, and the rise of megacities," *The Journal of Development Studies*, vol. 56, pp. 1663–1682, 9 2020.

[17] S. P. Nathaniel, "Natural resources, urbanisation, economic growth and the ecological footprint in south africa: The moderating role of human capital," *Quaestiones Geographicae*, vol. 40, pp. 63–76, 6 2021.

[18] K. H. Zhang, "What explains china's rising urbanisation in the reform era?," *http://dx.doi.org/10.1080/0042098022000033881*, vol. 39, pp. 2301–2315, 11 2002.

[19] L. Bertinelli and E. Strobl, "Urbanisation, urban concentration and economic development," *Urban Studies*, vol. 44, pp. 2499–2510, 12 2007.

[20] I. F. U. Muzayanah, H. H. Lean, D. Hartono, K. D. Indraswari, and R. Partama, "Population density and energy consumption: A study in indonesian provinces," *Heliyon*, vol. 8, p. e10634, 9 2022.

[21] Q. Wang and X. Yang, "Urbanization impact on residential energy consumption in china: the roles of income, urbanization level, and urban density," *Environmental Science and Pollution Research*, vol. 26, pp. 3542–3555, 2 2019.

[22] G. Strbac, "Demand side management: Benefits and challenges," *Energy Policy*, vol. 36, pp. 4419–4426, 12 2008.

[23] Q. Wu, L. Wang, and H. Cheng, "Research of tou power price based on multi-objective optimization of dsm and costs of power consumers," *Proceedings of the 2004 IEEE International Conference on Electric Utility Deregulation, Restructuring and Power Technologies (DRPT2004)*, vol. 1, pp. 343–348, 2004.

[24] M. Jarnut, R. Smolenski, G. Benysek, and J. Bojarski, "A comparative study on dsm services employing distributed energy resources: Energy storage and direct load control," *Proceedings of the 2017 6th International Conference on Electrical Engineering and Informatics: Sustainable Society Through Digital Innovation, ICEEI 2017*, vol. 2017-November, pp. 1–5, 7 2017.

[25] M. L. Nicolson, M. J. Fell, and G. M. Huebner, "Consumer demand for time of use electricity tariffs: A systematized review of the empirical evidence," *Renewable and Sustainable Energy Reviews*, vol. 97, pp. 276–289, 12 2018.

[26] E. V. Hobman, E. R. Frederiks, K. Stenner, and S. Meikle, "Uptake and usage of cost-reflective electricity pricing: Insights from psychology and behavioural economics," *Renewable and Sustainable Energy Reviews*, vol. 57, pp. 455–467, 5 2016.

[27] P. Grünewald, E. McKenna, and M. Thomson, "Keep it simple: time-of-use tariffs in high-wind scenarios," *IET Renewable Power Generation*, vol. 9, pp. 176–183, 3 2015.

[28] J. Torriti, "Price-based demand side management: Assessing the impacts of time-of-use tariffs on residential electricity demand and peak shifting in northern italy," *Energy*, vol. 44, pp. 576–583, 8 2012.

[29] K. Burns and B. Mountain, "Do households respond to time-of-use tariffs? evidence from australia," *Energy Economics*, vol. 95, p. 105070, 3 2021.

[30] T. Yunusov and J. Torriti, "Distributional effects of time of use tariffs based on electricity demand and time use," *Energy Policy*, vol. 156, p. 112412, 9 2021.

[31] K. Herter, "Residential implementation of critical-peak pricing of electricity," *Energy Policy*, vol. 35, pp. 2121–2130, 4 2007.

[32] M. A. Crew, C. S. Fernando, and P. R. Kleindorfer, "The theory of peak-load pricing: A survey," *Journal of Regulatory Economics*, vol. 8, pp. 215–248, 11 1995.

[33] K. Herter, P. McAuliffe, and A. Rosenfeld, "An exploratory analysis of california residential customer response to critical peak pricing of electricity," *Energy*, vol. 32, pp. 25–34, 1 2007.

[34] T. Kato, A. Tokuhara, Y. Ushifusa, A. Sakurai, K. Aramaki, and F. Maruyama, "Consumer responses to critical peak pricing: Impacts of maximum electricity-saving behavior," *The Electricity Journal*, vol. 29, pp. 12–19, 3 2016.

[35] A. Faruqui and S. Sergici, "Household response to dynamic pricing of electricity: A survey of 15 experiments," *Journal of Regulatory Economics*, vol. 38, pp. 193–225, 8 2010.

[36] E. Chueca, M. Weiss, R. Celaya, P. Ravillard, B. Ortega, M. T. Tolmasquim, and M. Hallack, "Early adopters of residential solar pv distributed generation: Evidence from brazil, chile and mexico," *Energy for Sustainable Development*, vol. 76, p. 101284, 10 2023.

[37] P. Paudyal, F. Ding, S. Ghosh, M. Baggu, M. Symko-Davies, C. Bilby, and B. Hannegan, "The impact of behind-the-meter heterogeneous distributed energy resources on distribution grids," *Conference Record of the IEEE Photovoltaic Specialists Conference*, vol. 2020-June, pp. 0857–0862, 6 2020.

[38] F. Farzan, F. Farzan, M. A. Jafari, and J. Gong, "Integration of demand dynamics and investment decisions on distributed energy resources," *IEEE Transactions on Smart Grid*, vol. 7, pp. 1886–1895, 7 2016.

[39] M. A. A. Pedrasa, T. D. Spooner, and I. F. MacGill, "Coordinated scheduling of residential distributed energy resources to optimize smart home energy services," *IEEE Transactions on Smart Grid*, vol. 1, pp. 134–143, 9 2010.

[40] A. Omu, R. Choudhary, and A. Boies, "Distributed energy resource system optimisation using mixed integer linear programming," *Energy Policy*, vol. 61, pp. 249–266, 10 2013.

[41] M. Neukomm, V. Nubbe, and R. Fares, "Grid-interactive efficient buildings technical report series: Overview of research challenges and gaps," 12 2019.

[42] K. N. Simani, Y. O. Genga, and Y.-C. J. Yen, "Using LSTM to perform load predictions for grid-interactive buildings," *SAIEE Africa Research Journal*, vol. 115, 2024.

[43] Y. Liu, L. Xiao, G. Yao, and S. Bu, "Pricing-based demand response for a smart home with various types of household appliances considering customer satisfaction," *IEEE Access*, vol. 7, pp. 86463–86472, 2019.

[44] X. Xue, S. Wang, Y. Sun, and F. Xiao, "An interactive building power demand management strategy for facilitating smart grid optimization," *Applied Energy*, vol. 116, pp. 297–310, 3 2014.

[45] D. Liu, Y. Xu, Q. Wei, and X. Liu, "Residential energy scheduling for variable weather solar energy based on adaptive dynamic programming," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, pp. 36–46, 1 2018.

[46] Z. Pooranian, J. H. Abawajy, P. Vinod, and M. Conti, "Scheduling distributed energy resource operation and daily power consumption for a smart building to optimize economic and environmental parameters," *Energies 2018, Vol. 11, Page 1348*, vol. 11, p. 1348, 5 2018.

[47] E. A. Soto, L. B. Bosman, E. Wollega, and W. D. Leon-Salas, "Peer-to-peer energy trading: A review of the literature," *Applied Energy*, vol. 283, p. 116268, 2 2021.

[48] M. S. Nizami, M. J. Hossain, K. Mahmud, and J. Ravishankar, "Energy cost optimization and der scheduling for unified energy management system of residential neighborhood," *Proceedings - 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe, EEEIC/I and CPS Europe 2018*, 10 2018.

[49] C. Long, J. Wu, C. Zhang, L. Thomas, M. Cheng, and N. Jenkins, "Peer-to-peer energy trading in a community microgrid," *IEEE Power and Energy Society General Meeting*, vol. 2018-January, pp. 1–5, 1 2018.

[50] H. Acaroglu and F. P. G. Márquez, "Comprehensive review on electricity market price and load forecasting based on wind energy," *Energies 2021, Vol. 14, Page 7473*, vol. 14, p. 7473, 11 2021.

[51] W. Ma, S. Fang, G. Liu, and R. Zhou, "Modeling of district load forecasting for distributed energy system," *Applied Energy*, vol. 204, pp. 181–205, 10 2017.

[52] H. G. Hassan, A. A. Shahin, and I. E. Ziedan, "Energy consumption forecast in peer to peer energy trading," *SN Applied Sciences*, vol. 5, pp. 1–10, 8 2023.

[53] Y. Iwafune, Y. Yagita, T. Ikegami, and K. Ogimoto, "Short-term forecasting of residential building load for distributed energy management," *ENERGYCON 2014 - IEEE International Energy Conference*, pp. 1197–1204, 2014.

[54] L. Zhao, T. Zhang, X. Peng, and X. Zhang, "A novel long-term power forecasting based smart grid hybrid energy storage system optimal sizing method considering uncertainties," *Information Sciences*, vol. 610, pp. 326–344, 9 2022.

[55] H. M. Al-Hamadi and S. A. Soliman, "Long-term/mid-term electric load forecasting based on short-term correlation and annual growth," *Electric Power Systems Research*, vol. 74, pp. 353–361, 6 2005.

[56] B. Kermanshahi and H. Iwamiya, "Up to year 2020 load forecasting using neural nets," *International Journal of Electrical Power & Energy Systems*, vol. 24, pp. 789–797, 11 2002.

[57] I. Badran, H. El-Zayyat, and G. Halasa, "Short-term and medium-term load forecasting for jordan's power system," *American Journal of Applied Sciences*, vol. 5, pp. 763–768, 2008.

[58] W. Charyloniuk and M. S. Chen, "Very short-term load forecasting using artificial neural networks," *IEEE Transactions on Power Systems*, vol. 15, pp. 263–268, 2 2000.

[59] M. Ghiassi, D. K. Zimbra, and H. Saidane, "Medium term system load forecasting with a dynamic artificial neural network model," *Electric Power Systems Research*, vol. 76, pp. 302–316, 3 2006.

[60] H. L. Willis, H. N. Tram, and G. B. Rackliffe, "Short-range load forecasting for distribution system planning – an improved method for extrapolating feeder load growth," *IEEE Transactions on Power Systems*, vol. 7, pp. 1306–1312, 1992.

[61] N. Cetinkaya, "Long-term electrical load forecasting based on economic and demographic data for turkey," *CINTI 2013 - 14th IEEE International Symposium on Computational Intelligence and Informatics, Proceedings*, pp. 219–223, 2013.

[62] A. K. Singh, Ibraheem, S. Khatoon, M. Muazzam, and D. K. Chaturvedi, "Load forecasting techniques and methodologies: A review," *ICPCES 2012 - 2012 2nd International Conference on Power, Control and Embedded Systems*, 2012.

[63] G. R. Esteves, B. Q. Bastos, F. L. Cyrino, R. F. Calili, and R. C. Souza, "Long term electricity forecast: A systematic review," *Procedia Computer Science*, vol. 55, pp. 549–558, 1 2015.

[64] H. K. Alfares and M. Nazeeruddin, "Electric load forecasting: Literature survey and classification of methods," *International Journal of Systems Science*, vol. 33, pp. 23–34, 1 2002.

[65] M. A. Mahmud and M. A. Mahmud, "Isolated area load forecasting using linear regression analysis: Practical approach," *Energy and Power Engineering*, vol. 3, pp. 547–550, 9 2011.

[66] E. Chodakowska, J. Nazarko, and Łukasz Nazarko, "Arima models in electrical load forecasting and their robustness to noise," *Energies 2021, Vol. 14, Page 7952*, vol. 14, p. 7952, 11 2021.

[67] G. Juberias, R. Yunta, J. G. Moreno, and C. Mendivil, "New arima model for hourly load forecasting," *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, vol. 1, pp. 314–319, 1999.

[68] C. Cheng, A. Sa-Ngasoongsong, O. Beyca, T. Le, H. Yang, Z. Kong, and S. T. Bukkapatnam, "Time series forecasting for nonlinear and non-stationary processes: a review and comparative study," *IIE Transactions*, vol. 47, pp. 1053–1071, 10 2015.

[69] K. Metaxiotis, A. Kagiannas, D. Askounis, and J. Psarras, "Artificial intelligence in short term electric load forecasting: a state-of-the-art survey for the researcher," *Energy Conversion and Management*, vol. 44, pp. 1525–1534, 6 2003.

[70] A. S. Ahmad, M. Y. Hassan, M. P. Abdullah, H. A. Rahman, F. Hussin, H. Abdullah, and R. Saidur, "A review on applications of ann and svm for building electrical energy consumption forecasting," *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 102–109, 5 2014.

[71] N. Ayub, M. Irfan, M. Awais, U. Ali, T. Ali, M. Hamdi, A. Alghamdi, and F. Muhammad, "Big data analytics for short and medium-term electricity load forecasting using an ai techniques ensembler," *Energies 2020, Vol. 13, Page 5193*, vol. 13, p. 5193, 10 2020.

[72] B. Schölkopf, "Svms - a practical consequence of learning theory," *IEEE Intelligent Systems and Their Applications*, vol. 13, pp. 18–21, 7 1998.

[73] G. Mitchell, S. Bahadoorsingh, N. Ramsamooj, and C. Sharma, "A comparison of artificial neural networks and support vector machines for short-term load forecasting using various load types," *2017 IEEE Manchester PowerTech, Powertech 2017*, 7 2017.

[74] Y. Fu, Z. Li, H. Zhang, and P. Xu, "Using support vector machine to predict next day electricity load of public buildings with sub-metering devices," *Procedia Engineering*, vol. 121, pp. 1016–1022, 1 2015.

[75] J. J. Hopfield, "Artificial neural networks," *IEEE Circuits and Devices Magazine*, vol. 4, pp. 3–10, 1988.

[76] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, pp. 943–947, 11 2015.

[77] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, "Recent advances in recurrent neural networks," 12 2017.

[78] L. Hernandez, C. Baladrón, J. M. Aguiar, B. Carro, A. J. Sanchez-Esguevillas, and J. Lloret, "Short-term load forecasting for microgrids based on artificial neural networks," *Energies*, vol. 6, pp. 1385–1408, 2013.

[79] S. Khan, N. Javaid, A. Chand, A. B. M. Khan, F. Rashid, and I. U. Afridi, "Electricity load forecasting for each day of week using deep cnn," *Advances in Intelligent Systems and Computing*, vol. 927, pp. 1107–1119, 2019.

[80] N. Shabbir, R. Amadiahangar, H. A. Raja, L. Kutt, and A. Rosin, "Residential load forecasting using recurrent neural networks," *Proceedings - 2020 IEEE 14th International Conference on Compatibility, Power Electronics and Power Engineering, CPE-POWERENG 2020*, pp. 478–481, 7 2020.

[81] B. Yildiz, J. I. Bilbao, and A. B. Sproul, "A review and analysis of regression and machine learning models on commercial building electricity load forecasting," *Renewable and Sustainable Energy Reviews*, vol. 73, pp. 1104–1122, 6 2017.

[82] S. V. Oprea and A. Bara, "Machine learning algorithms for short-term load forecast in residential buildings using smart meters, sensors and big data solutions," *IEEE Access*, vol. 7, pp. 177874–177889, 2019.

[83] L. Zhang, J. Wen, Y. Li, J. Chen, Y. Ye, Y. Fu, and W. Livingood, "A review of machine learning in building load prediction," *Applied Energy*, vol. 285, p. 116452, 3 2021.

[84] M. Sajjad, Z. A. Khan, A. Ullah, T. Hussain, W. Ullah, M. Y. Lee, and S. W. Baik, "A novel cnn-gru-based hybrid approach for short-term residential load forecasting," *IEEE Access*, vol. 8, pp. 143759–143768, 2020.

[85] L. Wen, K. Zhou, and S. Yang, "Load demand forecasting of residential buildings using a deep learning model," *Electric Power Systems Research*, vol. 179, p. 106073, 2 2020.

[86] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting-a novel pooling deep rnn," *IEEE Transactions on Smart Grid*, vol. 9, pp. 5271–5280, 9 2018.

[87] C. Kuster, Y. Rezgui, and M. Mourshed, "Electrical load forecasting models: A critical systematic review," *Sustainable Cities and Society*, vol. 35, pp. 257–270, 11 2017.

[88] A. Azeem, I. Ismail, S. M. Jameel, and V. R. Harindran, "Electrical load forecasting models for different generation modalities: A review," *IEEE Access*, vol. 9, pp. 142239–142263, 2021.

[89] A. Kag, Z. Zhang, and V. Saligrama, "Rnns incrementally evolving on an equilibrium manifold: A panacea for vanishing and exploding gradients?," 9 2019.

[90] T. Hegazy, P. Fazio, and O. Moselhi, "Developing practical neural network applications using back-propagation," *Computer-Aided Civil and Infrastructure Engineering*, vol. 9, pp. 145–159, 3 1994.

[91] R. C. Staudemeyer and E. R. Morris, "Understanding lstm – a tutorial into long short-term memory recurrent neural networks," 9 2019.

[92] A. Farzad, H. Mashayekhi, and H. Hassanpour, "A comparative performance analysis of different activation functions in lstm networks for classification," *Neural Computing and Applications*, vol. 31, pp. 2507–2521, 7 2019.

[93] J. Luo, Z. Zhang, Y. Fu, and F. Rao, "Time series prediction of covid-19 transmission in america using lstm and xgboost algorithms," *Results in Physics*, vol. 27, 8 2021.

[94] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," 5 2015.

[95] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A comparison of arima and lstm in forecasting time series," pp. 1394–1401, Institute of Electrical and Electronics Engineers Inc., 1 2019.

[96] "Multivariate lstm-fcns for time series classification," *Neural Networks*, vol. 116, pp. 237–245, 8 2019.

[97] T. Xiao and F. You, "Physically consistent deep learning-based day-ahead energy dispatching and thermal comfort control for grid-interactive communities," *Applied Energy*, vol. 353, p. 122133, 1 2024.

[98] J. G. Kim and B. Lee, "Automatic p2p energy trading model based on reinforcement learning using long short-term delayed reward," *Energies 2020, Vol. 13, Page 5359*, vol. 13, p. 5359, 10 2020.

[99] G. Hebrail and A. Berard, "Individual Household Electric Power Consumption." UCI Machine Learning Repository, 2012. DOI: https://doi.org/10.24432/C58K54.

[100] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, pp. 841–851, 1 2019.

[101] B. S. Kwon, R. J. Park, and K. B. Song, "Short-term load forecasting based on deep neural networks using LSTM layer," *Journal of Electrical Engineering and Technology*, vol. 15, pp. 1501–1509, 7 2020.

[102] S. H. Rafi, N. 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. 32436–32448, 2021.

[103] M. Decuyper, M. Stockhoff, S. Vandenberghe, al, and X. Ying, "An overview of overfitting and its solutions," *Journal of Physics: Conference Series*, vol. 1168, p. 022022, 2 2019.

[104] V. Mitra and H. Franco, "Coping with unseen data conditions: Investigating neural net architectures, robust features, and information fusion for robust speech recognition," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 08-12-September-2016, pp. 3783–3787, 2016.

[105] H. Rafiq, X. Shi, H. Zhang, H. Li, M. K. Ochani, and A. A. Shah, "Generalizability improvement of deep learning-based non-intrusive load monitoring system using data augmentation," *IEEE Transactions on Smart Grid*, vol. 12, pp. 3265–3277, 7 2021.

[106] R. L. de C. Costa, "Convolutional-lstm networks and generalization in forecasting of household photovoltaic generation," *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105458, 11 2022.

[107] J. Chen and D. Wang, "Long short-term memory for speaker generalization in supervised speech separation," *The Journal of the Acoustical Society of America*, vol. 141, pp. 4705–4714, 6 2017.

[108] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *The Journal of the Acoustical Society of America*, vol. 139, pp. 2604–2612, 5 2016.

[109] M. Wen, Z. Yu, W. Li, S. Luo, Y. Zhong, and C. Changqing, "Short-term load forecasting based on feature mining and deep learning of big data of user electricity consumption," *AIP Advances*, vol. 13, p. 125315, 12 2023.

[110] Z. Hamad and I. Abdulrahman, "Deep learning-based load forecasting considering data reshaping using matlab-simulink," *International Journal of Energy and Environmental Engineering*, vol. 13, p. 853, 6 2022.

[111] M. Ghofrani, M. Hassanzadeh, M. Etezadi-Amoli, and M. S. Fadali, "Smart meter based short-term load forecasting for residential customers," *NAPS 2011 - 43rd North American Power Symposium*, 2011.

[112] J. W. Taylor and M. B. Roberts, "Forecasting frequency-corrected electricity demand to support frequency control," *IEEE Transactions on Power Systems*, vol. 31, pp. 1925–1932, 5 2016.

[113] J. Huchtkoetter and A. Reinhardt, "A study on the impact of data sampling rates on load signature event detection," *Energy Informatics*, vol. 2, pp. 1–12, 9 2019.

[114] A. Durand, S. Hirzel, C. Rohde, M. Gebele, C. Lopes, E. Olsson, and R. Barkhausen, "Electric kettles: An assessment of energy-saving potentials for policy making in the European Union," *Sustainability*, vol. 14, no. 20, 2022.

[115] D. Murray, J. Liao, L. Stankovic, and V. Stankovic, "Understanding usage patterns of electric kettle and energy saving potential," *Applied Energy*, vol. 171, pp. 231–242, 2016.

[116] S. Matsumoto, K. Mizobuchi, and S. Managi, "Household energy consumption," *Environmental Economics and Policy Studies*, vol. 24, pp. 1–5, 1 2022.

[117] E. R. Frederiks, K. Stenner, and E. V. Hobman, "Household energy use: Applying behavioural economics to understand consumer decision-making and behaviour," *Renewable and Sustainable Energy Reviews*, vol. 41, pp. 1385–1394, 1 2015.

[118] R. Kowsari and H. Zerriffi, "Three dimensional energy profile:: A conceptual framework for assessing household energy use," *Energy Policy*, vol. 39, pp. 7505–7517, 12 2011.

[119] N. C. Iraganaboina and N. Eluru, "An examination of factors affecting residential energy consumption using a multiple discrete continuous approach," *Energy and Buildings*, vol. 240, p. 110934, 6 2021.

[120] A. Najmi, H. S. G, and A. Keramati, "Energy consumption in the residential sector: a study on critical factors," *International Journal of Sustainable Energy*, vol. 35, pp. 645–663, 8 2016.

[121] R. V. Jones, A. Fuertes, and K. J. Lomas, "The socio-economic, dwelling and appliance related factors affecting electricity consumption in domestic

buildings," *Renewable and Sustainable Energy Reviews*, vol. 43, pp. 901–917, 3 2015.

[122] R. R. Andridge and R. J. Little, "A review of hot deck imputation for survey non-response," *International Statistical Review*, vol. 78, pp. 40–64, 4 2010.

[123] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *Journal of Big Data 2021 8:1*, vol. 8, pp. 1–37, 10 2021.

[124] R. R. Andridge and R. J. Little, "The use of sample weights in hot deck imputation," *Journal of official statistics*, vol. 25, p. 21, 3 2009.

[125] D. Joenssen and U. Bankhofer, *Hot Deck Methods for Imputing Missing Data*, p. 63–75. 07 2012.

[126] P. Muhammad Ali and R. Faraj, "Data normalization and standardization: A technical report," 01 2014.

[127] L. Djordjević, M. I. Jordović-Pavlović, M. Ćojbašić, S. P. Galović, M. N. Popović, M. V. Nešić, and D. D. Markushev, "Influence of data scaling and normalization on overall neural network performances in photoacoustics," *Optical and Quantum Electronics*, vol. 54, pp. 1–14, 8 2022.

[128] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, 12 2020.

[129] A. A. Gorodetsky and J. D. Jakeman, "Gradient-based optimization for regression in the functional tensor-train format," *Journal of Computational Physics*, vol. 374, pp. 1219–1238, 12 2018.

[130] W. Ma, X. Liu, J. Duan, S. Peng, E. V. Lieres, L. L. Gunnell, K. Manwaring, X. Lu, J. Reynolds, J. Vienna, and J. Hedengren, "Machine learning with gradient-based optimization of nuclear waste vitrification with uncertainties and constraints," *Processes 2022, Vol. 10, Page 2365*, vol. 10, p. 2365, 11 2022.

[131] K. Kong, G. Li, M. Ding, Z. Wu, C. Zhu, B. Ghanem, G. Taylor, and T. Goldstein, "Robust optimization as data augmentation for large-scale graphs," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2022-June, pp. 60–69, 10 2020.

[132] A. Bird, P. Jedlicka, and H. Cuntz, "Dendritic normalisation improves learning in sparsely connected artificial neural networks," *PLOS Computational Biology*, vol. 17, p. e1009202, 8 2021.

[133] A. F. M. S. Saif, A. Garba, J. Awwalu, H. Arshad, and L. Q. Zakaria, "Performance comparison of min-max normalisation on frontal face detection using haar classifiers," *Pertanika Journal of Science and Technology*, vol. 25(S), 06 2017.

[134] T. J. Sefara, "The effects of normalisation methods on speech emotion recognition," *Proceedings - 2019 International Multidisciplinary Information Technology and Engineering Conference, IMITEC 2019*, 11 2019.

[135] I. Niño-Adan, I. Landa-Torres, E. Portillo, and D. Manjarres, "Influence of statistical feature normalisation methods on k-nearest neighbours and k-means in the context of industry 4.0," *Engineering Applications of Artificial Intelligence*, vol. 111, p. 104807, 5 2022.

[136] S. Gopal, K. Patro, and K. K. Sahu, "Normalization: A preprocessing stage," *IARJSET*, pp. 20–22, 3 2015.

[137] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48 of *Proceedings of Machine Learning Research*, pp. 1225–1234, PMLR, 6 2016.

[138] A. Nespoli, E. Ogliari, S. Pretto, M. Gavazzeni, S. Vigani, and F. Paccanelli, "Data quality analysis in day-ahead load forecast by means of lstm," *Proceedings - 2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe, EEEIC / I and CPS Europe 2020*, 6 2020.

[139] A. Nespoli, E. Ogliari, S. Pretto, M. Gavazzeni, S. Vigani, and F. Paccanelli, "Electrical load forecast by means of lstm: The impact of data quality," *Forecasting 2021, Vol. 3, Pages 91-101*, vol. 3, pp. 91–101, 2 2021.

[140] C. B. Sahin and B. Diri, "Robust feature selection with lstm recurrent neural networks for artificial immune recognition system," *IEEE Access*, vol. 7, pp. 24165–24178, 2019.

[141] W. Li and D. M. Becker, "Day-ahead electricity price prediction applying hybrid models of lstm-based deep learning methods and feature selection algorithms under consideration of market coupling," *Energy*, vol. 237, p. 121543, 12 2021.

[142] C. Tao, J. Lu, J. Lang, X. Peng, K. Cheng, and S. Duan, "Short-term forecasting of photovoltaic power generation based on feature selection and bias compensation–lstm network," *Energies 2021, Vol. 14, Page 3086*, vol. 14, p. 3086, 5 2021.

[143] N. Pai and V. Ilango, "Lstm neural network model with feature selection for financial time series prediction," *Proceedings of the 4th International Conference on IoT in Social, Mobile, Analytics and Cloud, ISMAC 2020*, pp. 672–677, 10 2020.

[144] X. Dai, J. Liu, and Y. Li, "A recurrent neural network using historical data to predict time series indoor pm2.5 concentrations for residential buildings," *Indoor Air*, vol. 31, pp. 1228–1237, 7 2021.

[145] T. N. Minh and L. Wolters, "Using historical data to predict application runtimes on backfilling parallel systems," *Proceedings of the 18th Euromicro Conference on Parallel, Distributed and Network-Based Processing, PDP 2010*, pp. 246–252, 2010.

[146] S. Figlewski, "Forecasting volatility using historical data," 5 1994.

[147] I. Richardson, M. Thomson, D. Infield, and C. Clifford, "Domestic electricity use: A high-resolution energy demand model," *Energy and Buildings*, vol. 42, pp. 1878–1887, 10 2010.

[148] G. K. Tso and K. K. Yau, "A study of domestic energy usage patterns in hong kong," *Energy*, vol. 28, pp. 1671–1682, 12 2003.

[149] Y. G. Yohanis, "Domestic energy use and householders' energy behaviour," *Energy Policy*, vol. 41, pp. 654–665, 2 2012.

[150] K. K. Dobbin and R. M. Simon, "Optimally splitting cases for training and testing high dimensional classifiers," *BMC Medical Genomics*, vol. 4, pp. 1–8, 4 2011.

[151] I. M. Guyon, "A scaling law for the validation-set training-set size ratio," 1997.

[152] Y. S. Chen, P. P. Chong, and M. Y. Tong, "Mathematical and computer modelling of the pareto principle," *Mathematical and Computer Modelling*, vol. 19, pp. 61–80, 5 1994.

[153] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[154] F. Chollet *et al.*, "Keras: Deep learning for humans." `https://keras.io`, 2015.

[155] L. Yu, J. Qu, F. Gao, and Y. Tian, "A novel hierarchical algorithm for bearing fault diagnosis based on stacked lstm," *Shock and Vibration*, vol. 2019, 2019.

[156] X. Du, H. Zhang, H. V. Nguyen, and Z. Han, "Stacked lstm deep learning model for traffic prediction in vehicle-to-vehicle communication," *IEEE Vehicular Technology Conference*, vol. 2017-September, pp. 1–5, 7 2017.

[157] K. Vijayaprabakaran and K. Sathiyamurthy, "Towards activation function search for long short-term model network: A differential evolution based approach," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, pp. 2637–2650, 6 2022.

[158] M. Pirhooshyaran and L. V. Snyder, "Multivariate, multistep forecasting, reconstruction and feature selection of ocean waves via recurrent and sequence-to-sequence networks," 6 2019.

[159] T. M. Breuel, "Benchmarking of lstm networks," 8 2015.

[160] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 12 2014.

[161] S. Jiang and Y. Chen, "Hand gesture recognition by using 3dcnn and lstm with adam optimizer," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10735 LNCS, pp. 743–753, 2018.

[162] Z. Chang, Y. Zhang, and W. Chen, "Effective adam-optimized lstm neural network for electricity price forecasting," *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, vol. 2018-November, pp. 245–248, 7 2018.

[163] N. Bakhashwain and A. Sagheer, "Online tuning of hyperparameters in deep lstm for time series applications," *International Journal of Intelligent Engineering and Systems*, vol. 14, p. 2021.

[164] R. Solgi, H. A. Loáiciga, and M. Kram, "Long short-term memory neural network (lstm-nn) for aquifer level time series forecasting using in-situ piezometric observations," *Journal of Hydrology*, vol. 601, p. 126800, 10 2021.

[165] N. Saurabh, "Lstm-rnn model to predict future stock prices using an efficient optimizer," *International Research Journal of Engineering and Technology*, 2020.

[166] Q. Tong, G. Liang, and J. Bi, "Calibrating the adaptive learning rate to improve convergence of adam," *Neurocomputing*, vol. 481, pp. 333–356, 4 2022.

[167] N. S. Keskar and R. Socher, "Improving generalization performance by switching from adam to sgd," 12 2017.

[168] Z. Chang, Y. Zhang, and W. Chen, "Electricity price prediction based on hybrid model of adam optimized lstm neural network and wavelet transform," *Energy*, vol. 187, p. 115804, 11 2019.

[169] K. K. Chandriah and R. V. Naraganahalli, "Rnn / lstm with modified adam optimizer in deep learning approach for automobile spare parts demand forecasting," *Multimedia Tools and Applications*, vol. 80, pp. 26145–26159, 7 2021.

[170] C. Yu, X. Qi, H. Ma, X. He, C. Wang, and Y. Zhao, "Llr: Learning learning rates by lstm for training neural networks," *Neurocomputing*, vol. 394, pp. 41–50, 6 2020.

[171] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 11 2017.

[172] I. Jindal, M. Nokleby, and X. Chen, "Learning deep networks from noisy labels with dropout regularization," pp. 967–972, 2 2017.

[173] A. Poernomo and D. K. Kang, "Biased dropout and crossmap dropout: Learning towards effective dropout regularization in convolutional neural network," *Neural Networks*, vol. 104, pp. 60–67, 8 2018.

[174] J. Billa, "Dropout approaches for lstm based speech recognition systems," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2018-April, pp. 5879–5883, 9 2018.

[175] V. Pham, T. Bluche, C. Kermorvant, and J. Louradour, "Dropout improves recurrent neural networks for handwriting recognition," *Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR*, vol. 2014-December, pp. 285–290, 12 2014.

[176] M. Sangiorgio and F. Dercole, "Robustness of lstm neural networks for multi-step forecasting of chaotic time series," *Chaos, Solitons & Fractals*, vol. 139, p. 110045, 10 2020.

[177] Y. Liu, D. Hou, J. Bao, and Y. Qi, "Multi-step ahead time series forecasting for different data patterns based on lstm recurrent neural network," *Proceedings - 2017 14th Web Information Systems and Applications Conference, WISA 2017*, vol. 2018-January, pp. 305–310, 7 2017.

[178] D. K. Roy, T. K. Sarkar, S. S. A. Kamar, T. Goswami, M. A. Muktadir, H. M. Al-Ghobari, A. Alataway, A. Z. Dewidar, A. A. El-Shafei, and M. A. Mattar, "Daily prediction and multi-step forward forecasting of reference evapotranspiration using lstm and bi-lstm models," *Agronomy 2022, Vol. 12, Page 594*, vol. 12, p. 594, 2 2022.

[179] G. Jiménez-Castillo, C. Rus-Casas, G. M. Tina, and F. J. Muñoz-Rodriguez, "Effects of smart meter time resolution when analyzing photovoltaic self-consumption system on a daily and annual basis," *Renewable Energy*, vol. 164, pp. 889–896, 2 2021.

[180] M. Liang, Y. Meng, N. Lu, D. Lubkeman, and A. Kling, "Hvac load disaggregation using low-resolution smart meter data," *2019 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference, ISGT 2019*, 2 2019.

[181] M. Masum, H. Shahriar, H. M. Haddad, and M. S. Alam, "R-lstm: Time series forecasting for covid-19 confirmed cases with lstmbased framework,"

Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020, pp. 1374–1379, 12 2020.

[182] F. Shahid, A. Zameer, and M. Muneeb, "Predictions for covid-19 with deep learning models of lstm, gru and bi-lstm," *Chaos, Solitons & Fractals*, vol. 140, p. 110212, 11 2020.

[183] W. Wei, H. Wu, and H. Ma, "An autoencoder and lstm-based traffic flow prediction method," *Sensors 2019, Vol. 19, Page 2946*, vol. 19, p. 2946, 7 2019.

[184] K. A. Althelaya, E. S. M. El-Alfy, and S. Mohammed, "Evaluation of bidirectional lstm for short and long-term stock market prediction," *2018 9th International Conference on Information and Communication Systems, ICICS 2018*, vol. 2018-January, pp. 151–156, 5 2018.

[185] C. H. Wu, C. C. Lu, Y. F. Ma, and R. S. Lu, "A new forecasting framework for bitcoin price with lstm," *IEEE International Conference on Data Mining Workshops, ICDMW*, vol. 2018-November, pp. 168–175, 7 2018.

[186] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, pp. 1–24, 7 2021.

[187] A. Gelman, B. Goodrich, J. Gabry, and A. Vehtari, "R-squared for bayesian regression models," *The American Statistician*, vol. 73, pp. 307–309, 7 2019.

[188] A. C. Cameron and F. A. Windmeijer, "R-squared measures for count data regression models with applications to health-care utilization," *Journal of Business & Economic Statistics*, vol. 14, pp. 209–220, 1996.

[189] M. A. Rodriaguez-Otero and E. O'Neill-Carrillo, "Efficient home appliances for a future dc residence," *2008 IEEE Energy 2030 Conference, ENERGY 2008*, 2008.

[190] F. Issi and O. Kaplan, "The determination of load profiles and power consumptions of home appliances," *Energies 2018, Vol. 11, Page 607*, vol. 11, p. 607, 3 2018.

[191] P. Faria, J. Spínola, and Z. Vale, "Distributed energy resources scheduling and aggregation in the context of demand response programs," *Energies 2018, Vol. 11, Page 1987*, vol. 11, p. 1987, 7 2018.

[192] M. K. Mishra and S. K. Parida, "A game theoretic approach for demand-side management using real-time variable peak pricing considering distributed energy resources," *IEEE Systems Journal*, vol. 16, pp. 144–154, 3 2022.