

# **Using Machine Learning and Deep Learning for Load disaggregation and Recognition of Activities in Household**

by

Theogene Bimenyimana

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in  
partial fulfillment of the requirements for the degree of

Master of Information Technology

in

Digital Media with Specialization in Data Science

Carleton University

Ottawa, Ontario

© 2020, Theogene Bimenyimana

## **Abstract**

Providing statistics about energy consumptions and accurate forecasts at appliances level can help energy clients and users to make informed decisions and reduce their electricity bills. This thesis presents a forecasting strategy that combines the benefits of load disaggregation, weather information, and ensemble learning to improve the accuracy of short-term load forecasts in a household.

This thesis used combinatorial optimization to extract signature features of appliances from aggregated house energy consumption. The ensemble learning based on long short-term memory and the random forecast was designed to learn from the estimated energy consumption of appliances and predict the household energy consumption for three days ahead.

The model was tested on a publicly available dataset, and the results showed that the load forecasting approach that considered the dynamic behaviors of home residents outperforms the accuracy of the benchmark forecasting methods that are trained with weather information or prior aggregated house power demand.

## **Acknowledgements**

Foremost, I would like to express my sincere gratitude to my supervisor, Dr. M. Omair Shafiq for the continuous support, encouragement, constructive critiques, the guidance he has given me through my two years of master's studies at Carleton University. Throughout this research, I learned from his advice and expertise to explore new ideas and techniques. His valuable comments and quick feedback at various stage of this research has helped me to wisely organize and complete my thesis in a time limit. Dr. Omair Shafiq has always been a teacher and supervisor who extends his availability from academic to non-academics' guidance. It was a privilege and an honor for me to have friendly discussions during every research meeting.

My thanks also extend to Dr. Ali Arya, and Dr. Bonaventure Karuta for their motivating pieces of advice that helped me working on exciting projects in Information Technology.

To Carleton University, I would like to express my sincere thanks for providing excellent learning materials, research conditions that helped me in my research development and completion.

I would like to thank my parents Faustin and Esperance, my uncle Evariste and aunt Mechtilde who gave me the values that helped me during this research. My cordial thanks to my beloved and best friend Feza, her encouragement, unconditional friendship, and support throughout these years.

Above all, Glory to the Lord, the source of knowledge and wisdom for His love and protection.

## Table of Contents

<b>CHAPTER1: INTRODUCTION AND BACKGROUND .....</b>	<b>1</b>
1.1. Introduction and Background.....	1
1.2 Motivation.....	3
1.3 Summary of gap analysis and problem statement .....	4
1.4 Summary of contributions (key contributions) .....	6
1.5 Structure of the thesis .....	7
<b>CHAPTER 2: LITERATURE REVIEW .....</b>	<b>8</b>
2.1. Review on Load disaggregation-based Datasets .....	8
2.2. Review on Existing Load Forecasting and Disaggregation Approaches.....	14
2.2.1. Review on Load forecasting approaches.....	14
2.2.2. Load Disaggregation based Approaches .....	20
<b>CHAPTER 3: DATASET DESCRIPTION .....</b>	<b>34</b>
3. 1 Gokagglers dataset for aggregated appliances energy prediction .....	34
3.1.1 Information in Gokagglers dataset.....	34
3.1.2 Data visualization of Gokagglers dataset .....	35
3.2. Reference Energy Disaggregation Dataset (REDD).....	38
3.2.1 Information in REDD dataset.....	38
3.2.2 Visualization of REDD dataset.....	39
3.3. Visualization of Almanac of minutely power datasets (AMPDs) .....	41
<b>CHAPTER 4: GAP ANALYSIS, RESEARCH QUESTION, AND PROBLEM STATEMENT .....</b>	<b>44</b>
4.1. Gap in Existing Load Disaggregation based Datasets .....	44
4.2. Gap in existing Load Forecasting and Disaggregation based Approaches .....	45
4.3. Research Questions.....	47
4.4. Problem Statement.....	48
4.5. Key contributions .....	49
<b>CHAPTER 5: EXPERIMENTS USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES FOR LOAD FORECAST AND DISAGGREGATION.....</b>	<b>51</b>
5.1 Load forecasting-based techniques.....	51

5.1.1 Experiment 1: Random Forest (Ensemble Learning), add bagging and boosting (AdaBoost, XGBoost, Gradient Boosting) .....	52
5.1.2 Experiment 2: Support Vector Machine (SVM) for load forecasting.....	57
5.1.3 Performance comparison of load forecasting approaches .....	58
5.2 Load disaggregation.....	59
5.2.1 Experiment 1: Combinatorial Optimization (CO) for Load Disaggregation ...	59
5.2.2 Experiment 2: Factorial Hidden Markov Model (FHMM) for Load Disaggregation.....	64
5.2.3 Experiment 3: Denoising Autoencoder (DAE) Neural Network for Load Disaggregation.....	67
5.2.4 Experiment 4: Recurrent Neural Network based Long Short-Term Memory (LSTM) for Load Disaggregation .....	70
5.2.5 Experiment 5: Gated Recurrent Unit (GRU) Neural Network for Load Disaggregation.....	72
5.2.6 Experiment 6: Convolution based Seq 2-point Neural Network for Load Disaggregation.....	75
5.2.7 Performance comparison of load disaggregation-based approaches .....	77
<b>CHAPTER 6: PROPOSED SOLUTION.....</b>	<b>80</b>
6.1. Combinatorial optimization-based load disaggregation .....	81
6.1.1 Data Preprocessing.....	82
6.1.2. Detailed technical design of Combinatorial optimization for load disaggregation .....	84
6.2. Load forecasting based on load disaggregation .....	89
6.2.1. Data Description .....	90
6.2.2. Stacking Modelling for Load Forecasting.....	92
<b>CHAPTER 7: EVALUATION AND RESULTS .....</b>	<b>100</b>
7.1 Functional evaluation.....	100
7.2 Efficiency evaluation .....	107
7.3 Complexity analysis.....	109
7.4 Comparison of results .....	111
<b>CHAPTER 8: CONCLUSION AND FUTURE WORK.....</b>	<b>116</b>
8.1. Conclusion.....	116
8.2. Future Work .....	118
<b>Bibliography.....</b>	<b>120</b>

## List of Figures

Figure 1.1:Patterns of energy consumption of four categories of appliances.....	2
Figure 3.1: Daily appliances power consumption.....	36
Figure 3.2: Weekly mean appliances power consumption per week .....	36
Figure 3.3: Appliances' weekly mean power consumption every month.....	37
Figure 3.4: Appliances hourly power consumption .....	37
Figure 3.5: Correlation between appliances consumption and in-door and outdoor features in Gokagglers dataset. ....	38
Figure 3.6: Aggregated and appliances energy consumption from house2 of REDD dataset. ....	40
Figure 3.7: Appliances energy consumption from house2 of REDD dataset.....	41
Figure 3.8: Microwave energy consumption from house2 of REDD dataset.....	41
Figure 3.9: Hourly aggregated house power demand in AMPds.....	42
Figure 3.10: Energy consumption of 7 important appliances in AMPds .....	43
Figure 5.1: Overview of load forecasting procedure.....	52
Figure 5.2:Hyperparameter tuning for random forest with gradient boosting method .....	53
Figure 5.3: Random forest predicted vs actual power demand between Mach 17 and March 21,2016. ....	55
Figure 5.4:Features importance in forecasting performance of GradientBoostingRegressor. ....	56
Figure 5.5: RF model for load forecast .....	56
Figure 5.6: SVR predicted vs actual power demand between May 01 and May 06,2016.....	57
Figure 5.7: Forecasting accuracy of SVR.....	58
Figure 5.8:Performance comparison for regression-based load forecasting approaches ..	59
Figure 5.9: Appliance power disaggregated from site meter with CO approach.....	63
Figure 5.10: F1 score obtained with CO for home 1 seen during training .....	64
Figure 5.11: Appliance power disaggregated from site meter with FHMM approach .....	66
Figure 5.12: F1 Score performance comparison between CO and FHMM approach.....	66
Figure 5.13: DAE estimated vs ground truth power demand of fridge and microwave in house1 .....	69
Figure 5.14: DAE disaggregation F1 score for different appliances.....	69
Figure 5.15: LSTM estimated vs ground truth power demand of fridge and microwave in house1 .....	72
Figure 5.16: LSTM disaggregation F1 score for different appliance.....	72
Figure 5.17: GRU estimated vs ground truth power demand of fridge and microwave in house1 .....	74
Figure 5.18: GRU disaggregation F1 score for different appliance.....	75
Figure 5.19: CNN Estimated vs ground truth power demand of fridge and microwave in house1 .....	77

Figure 5.20: F1 score comparison of load disaggregation-based approach tested on home1 whose data seen during training.....	79
Figure 6.1: Overall conceptual model of proposed household load forecast based on load disaggregation .....	81
Figure 6.2: Steps for appliances power estimation based on Combinatorial optimization	82
Figure 6.3: Process of heterogeneous ensemble learning based load forecasting .....	90
Figure 6.4: Stacking load forecasting model .....	93
Figure 6.5: Customized LSTM architecture with appliances power demand and weather data, adapted from [48].....	98
Figure 7.1: Proportion of appliance energy in the actual power demand from AMPDs dataset. ....	102
Figure 7.2: Proportion of appliance energy in the CO- predicted power demand from AMPDs dataset. ....	102
Figure 7.3: The disaggregated energy consumption of appliances with CO on AMPDs .	103
Figure 7.4: CO disaggregation F1 score for different appliances present in home 1 seen during AMPDs data training.....	104
Figure 7.5: The actual house power demand and predicted house total power demand by different forecasting models. ....	106
Figure 7.6: NRMSE comparison of load forecasting models based on load disaggregation .....	107
Figure 7.7: Computational time of load forecasting models based on load disaggregation .....	108
Figure 7.8: NRMSE error for load forecasting model based on number of input data(features).....	110
Figure 7.9: NRMSE error for load forecasting model based on forecasting length. ....	110
Figure 7.10: Computational time for load forecasting based on number of input data(features).....	111
Figure 7.11: F1 score comparison on disaggregating appliances' energy consumption present in AMPDs dataset using different load disaggregation-based approaches. ....	113
Figure 7.12: MAPE and RMSE comparison for different load forecasting approaches tested on different dataset .....	115

## List of Tables

Table 2.1:Details of publicly available datasets for electric energy disaggregation.....	14
Table 2.2: Comparison of load disaggregation-based Approaches.....	31
Table 3.1: Features information in Gokagglers dataset. Taken from [15] .....	34
Table 3.2: Features information in REDD dataset. Taken from [16].....	39
Table 5.1: RF model for load forecast.....	56
Table 5.2: SVM model performance for load forecast.....	58
Table 5.3: Performance comparison for regression-based load forecasting approaches ..	58
Table 5.4: CO energy disaggregation performance for home 1 seen during training.....	63
Table 5.5: FHMM energy disaggregation performance for home 1 seen during training	66
Table 5.6: DAE Energy disaggregation performance for home seen during training.....	69
Table 5.7: LSTM energy disaggregation performance analysis for home 1 seen during training.....	72
Table 5.8: GRU Energy disaggregation performance for home 1 seen during training. ..	74
Table 5.9: Conv sequence to point Energy disaggregation performance analysis for home seen during training. ....	77
Table 5.10: F1 score comparison of load disaggregation-based approach tested on home1 seen during training. ....	78
Table 6.1: Customized Pseudocode of customized CO based patterns similarity, adapted from [50].....	87
Table 6.2: The optimal parameters and search spaces of forecasting algorithms.....	95
Table 6.3: Customized pseudocode of RF-LSTM Ensemble for load forecasting, adapted from [26].....	99
Table 7.1:CO energy disaggregation performance in home 1 seen during AMPDs data training.....	104
Table7.2:Performance comparison of load forecasting models based on load disaggregation. ....	106
Table 7.3: Computation time of load forecasting models based on load disaggregation. ....	108
Table 7.4: Computation time of load forecasting model based on number of features in the training stage. ....	109
Table 7.5: F1 score comparison on disaggregating appliances' energy consumption present in AMPDs dataset using different load disaggregation-based approaches. ....	113
Table 7.6: MAPE and RMSE comparison for different load forecasting approaches tested on different dataset. ....	114

# CHAPTER1: INTRODUCTION AND BACKGROUND

We introduce and provide background on the load disaggregation and forecast in Section 1.1. Section 1.2 presents the motivation for the load disaggregation and forecast. Section 1.3 describes the gap in the existing load disaggregation and forecast as well as the problem statement. The key contributions of this thesis are presented in Section 1.4. Section 1.5 provides the structure of the entire thesis.

## **1.1. Introduction and Background**

In the past years, the energy management system has accommodated technologies that include smart appliances, electric billing, and advanced communications within the power system. The objectives of the available demand response schemes can be achieved if energy clients are actively encouraged to save energy and reduce their monthly electricity bill [1].

The appliances' energy use is a key factor that shows the electricity consumption behavior of home residents. The consumption behaviors are reflected in how and when home residents use home appliances. Appliances features are modeled to distinguish the active and inactive state of a given appliance as shown in figure 1.1. Based on appliances features, the appliances are classified into: (1) The appliances with binary states of energy consumption that can be observed when they are turned ON and OFF. Light is an example of an appliance that falls in this category. (2) The appliances with finite or multiple states of operation in which each state is associated with the corresponding energy consumption. The appliances with multiple states include the lamp, stove burners, etc. (3) The appliances with the infinite state that are characterized by a continuously changing energy

consumption. The washer-dryer is an example of an appliance that falls in this category and its power consumption keeps fluctuating in time of drying clothes in a laundry. (4) The always ON state types of appliances that are always consuming energy [2,3]. The fridge is an example of an appliance that falls in this category.

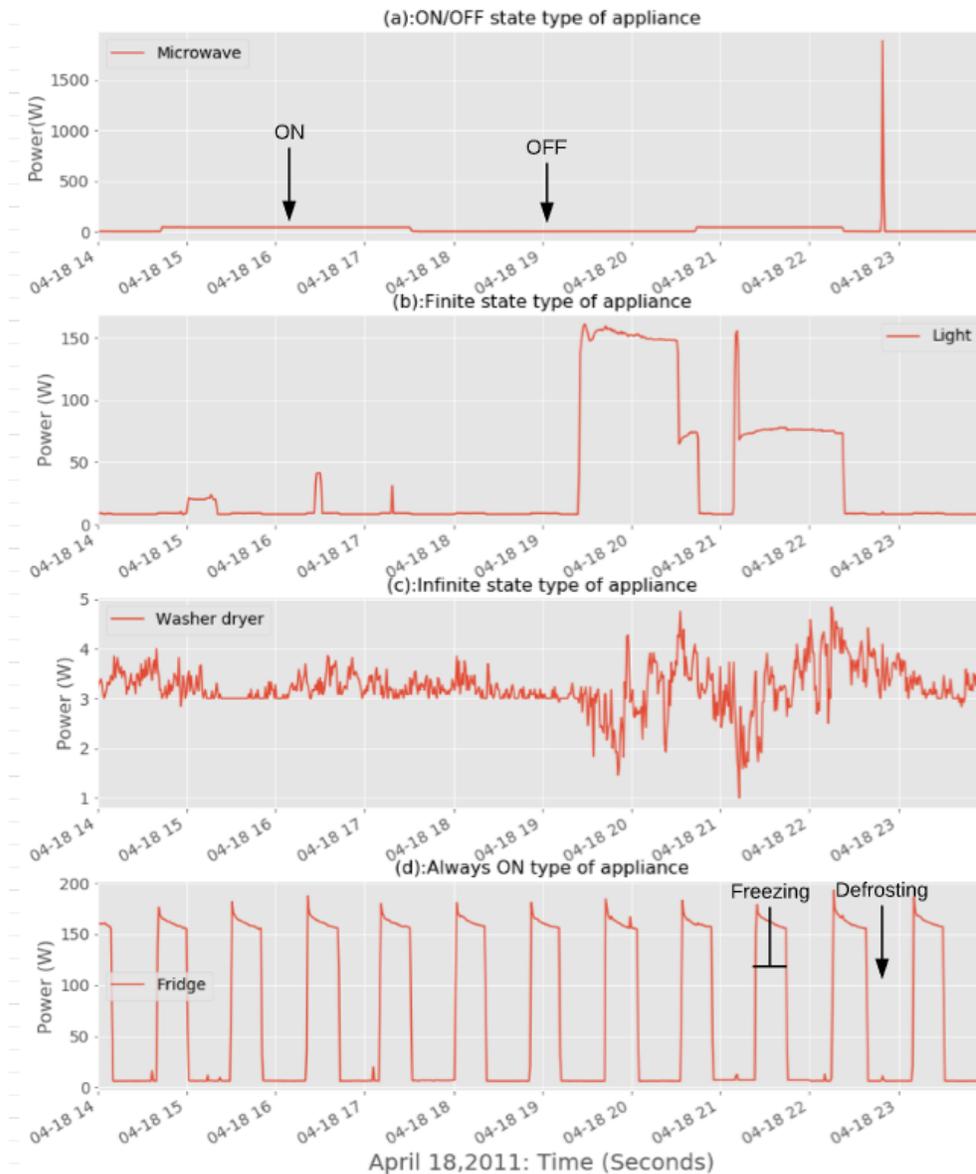


Figure 1.1:Patterns of energy consumption of four categories of appliances.

(a) ON/OFF type: Microwave;(b) finite or multistate type: Lighting bulb;(c) infinite or continuously varying signal: Washer dryer;(d) always on: Fridge

The load disaggregation is defined as a computation technique used to separate the energy consumption of individual home appliances from the aggregated home energy consumption recorded by a single smart meter in the house [4]. The load disaggregation which is known as nonintrusive load management (NILM) is aimed to provide energy feedback at appliances level without installing the energy monitors or sensor for each appliance.

The household's power demand constitutes a large portion of a complex distribution power system and this indicates that the accurate short term load forecast in a household can help energy clients and utility companies to contribute to energy saving and manage the burden of the entire grid [5,6].

## **1.2 Motivation**

Providing the energy consumption feedbacks in a short time domain (e.g daily) can help energy consumers to cut or reduce energy consumption and save money. Imagine the situation in which, the energy consumer got the following message: The system shows that “your washing machine has consumed 10 kWh for the last 7 days and this costs 17CAN\$. The estimated yearly consumption of a washing machine is 850 CAN\$.

It has been reported that providing energy feedback at the appliance level can help home residents to be aware of future energy consumption and save 5-12 % of their total energy bills. [7,8]. The smart meter technology that provides energy consumption at the house level has been worldwide adopted but it can not provide effective feedbacks than having a smart meter to each appliance present in a house. Moreover, installing the energy monitor on each appliance is more expensive and can cost 300-600\$ for each house [9].

In comparison with the cost of buying and installing the energy monitor on every appliance, the load disaggregation approach can achieve the personalized feedback scheme at the appliance level and save money.

The load forecast based on disaggregated appliances and weather information can reduce the forecast error and help energy consumer to progressively change their energy consumption behavior.

### **1.3 Summary of gap analysis and problem statement**

Research in load disaggregation has explored the potential of supervised and semi-supervised learning technologies for providing individualized energy feedback at the appliance level. In supervised learning, the prior appliances' data are required for the model training. The method is reported to achieve accurate results but computationally becomes complex when more appliances are included in the model training set. The optimization and probabilistic based approaches are examples of a supervised based load disaggregation approach. As supervised learning-based approaches are trained with sub-metered appliances information, the combinatorial optimization [6] and hidden Markov models [9] have reported achieving the high disaggregation accuracy for disaggregating the ON/OFF states of appliances.

In a semi-supervised learning-based approach, the model is designed to learn from the aggregated smart meter data or few appliance level data to disaggregate features of an individual appliance. Even if this type of learning has reported achieving less accuracy compared to supervised learning, the long short term memory (LSTM)[4] and sequence to point-based architectures[10] have reported succeeding with disaggregating the appliances

with complex features such as the washing machine. Deep neural network architectures are popular for load disaggregation because of their ability to disaggregate the complex signature features that represent the appliances in a multi-state of operation. In addition to this, the deep learning model can be trained once with appliances level data in one house and be used to disaggregated signal features of appliances in other houses which could help in achieving transfer learning or applicability.

Compared to the hidden Markov-based approaches, the combinatorial optimization can easily disaggregate the aggregated power demand by minimizing the mismatch between household total load and the sum of energy consumption of individual appliances that contributed to the total load. Moreover, the optimization approach is computationally simple compared to the sophisticated deep learning-based approaches.

Considering the variability and volatility of the household power demand, it is essential to develop the forecast model that considers the energy consumers' behavior and reduce the short-term load forecast. The forecasting accuracy of any load forecasting techniques is highly dependent on the preprocessing of input data. Therefore, data preprocessing based on load disaggregation can help in addressing the uncertainty of household power demand and achieving the accurate short-term load forecast.

The existing load forecast approaches are trained with (1) the prior time series total load to predict the future house energy consumption. This type of method is simply because the model is trained on the regular load patterns. (2) The load forecasting methods can also be trained with related variables to forecast future household power demand. For example, the model can learn from the meteorological weather parameters and predict future energy consumption in the household. (3) The third category is the load forecast based on data-

preprocessing in which data are preprocessed to separate the aggregated load with appliances energy consumption using NILM techniques and use them to forecast the house power demand[6,11].

The house energy consumption is affected with both in house dynamic use of appliances and external change of meteorological weather conditions. Most of the forecasting techniques are trained with either prior aggregated house energy consumption or meteorological weather conditions. Few load forecasting approaches are trained on appliances features extracted with load disaggregation [6,11]. To the best of our knowledge, none of the existing research has combined the benefits of appliances energy signal and external change of meteorological weather conditions for estimating the future house energy consumption. Therefore, it is essential to develop a load forecasting -based architecture that utilizes the information from appliances energy use and weather data.

#### **1.4 Summary of contributions (key contributions)**

In this thesis, we developed the household load forecast architecture that uses weather data and data extraction from load disaggregation to forecast the 3 days energy consumption of a small household. The combinatorial optimization approach was used to extract the appliances' energy consumption from the aggregated house power demand. The hybrid ensemble learning made with long short-term memory (LSTM) and random forest (RF) is constructed to forecast the house total power based on disaggregated appliances data and weather data.

The contributions of this thesis are summarized as follows:

(1) A comparative analysis of the latest machine learning and deep learning techniques applied to load disaggregation.

(2) We developed a forecasting method that combines the benefits of load disaggregation and meteorological weather conditions to tackle the uncertainty of household power demand and reduce the error in the load forecast.

(3) We compared the effectiveness of an LSTM-RF on forecasting the power demand of a small energy household with other forecasting methods.

### **1.5 Structure of the thesis**

This thesis is organized as follows: Chapter II describes the literature on load forecasting and disaggregation. Chapter III describes the NILM dataset description. Chapter IV provides the gap analysis, research questions, and problem statement. Chapter V gives the experiment results and comparison of load forecasting and disaggregation-based approaches. Chapter VI describes the proposed load forecast based on load disaggregation. Chapter VII presents the evaluation and results from the discussion. Finally, Chapter VII, the conclusions, and future work are deduced from the development of this paper.

## CHAPTER 2: LITERATURE REVIEW

This chapter discusses the related works to the objective of the thesis. Section 2.1. covers literature on datasets for load disaggregation; Section 2.2 presents the review of load disaggregation-based techniques as well as the existing techniques for load forecast.

### 2.1. Review on Load disaggregation-based Datasets

The practical load disaggregation uses both appliances and house level data for training and testing any machine learning and deep learning algorithm [12]. The smart meters can record the energy-related information, but the challenge in data preprocessing limits having a great number of NILM datasets on the public. The smart meters record a large amount of data which makes preprocessing difficult and expensive.

The major datasets for load disaggregation-based models include:

**Overview:** Rashid et al. [13] presented the Indian Buildings Energy consumption datasets (I- BLEND A) that comprise of 52 months of electrical energy with a one-minute sampling frequency. The dataset contains the energy parameters of 7 commercial and 7 residential buildings. In addition to power-related data, this dataset provides occupancy and weather datasets. The occupancy data were recorded every 10 minutes from each IIT Delhi campus building whereas weather data is obtained from the nearby weather station.

**Strengths:** The datasets contain data for both commercial and residential buildings which is important in comparing the energy consumption patterns between the commercial and residential buildings. Also, the fact that the datasets are recorded for long-duration makes it fit for deep learning. In addition to this energy data, providing the weather input parameters can help in evaluating the impact of ambient parameters to the load forecasting model.

**Weaknesses:** The datasets present the total electrical consumption and lack individual appliance consumption that is also required for training some load disaggregation-based approach in providing feedback at appliances-level.

**Overview:** Batra et al. [14] published the Indian data for ambient water and electricity sensing(iAWE,2014) that combines the sensor-measured energy consumption, water consumption, and ambient weather information for 3 homes in New Delhi, India. The data were recorded for 3 months (May-August 2013) duration.

**Strengths:** The advantages of the dataset are that it associated electrical appliances with metadata such as appliance name, age, mode of usage (e.g. air conditioner set temperature). These metadata details could provide insights to energy consumers and take decisions for saving electricity. The dataset contains weather-related information that is important for predicting the future load forecast as weather parameters are the key important causes of fluctuating energy consumption in the household.

**Weaknesses:** The data came with different sampling frequencies which complicate its use for load forecast. This indicates that the additional data preprocessing is required for matching the sampling of the dataset's information. Furthermore, the datasets were recorded for a short duration (73 days) which limits the evaluation of load disaggregation-based model in a long period.

**Overview:** Candanedo et al. [15] provided the combined appliances energy consumption, indoor weather data information for 1 house in Belgium, and outdoor weather information from the nearby meteorology station. The datasets cover recording for 4.5 months with 10 min sampling frequency.

**Strengths:** The data offers a mix of total appliances level energy consumption and weather input parameters which make it applicable for load forecasting. The facts that the datasets contain both in and outside door weather features are essential in determining the correlation and importance of the features.

**Weaknesses:** The dataset has records for a short duration (4.5 months) which limits investigating the importance of weather features to the appliance's consumption in a whole year season (Fall, Winter, Spring, and Summer).

**Overview:** Kolter et al. [16] made available the Reference Energy Disaggregation Datasets (REDD) that provides data for 6 USA houses. The data were recorded for 19 days with a sampling frequency of 15kHz (high-frequency data) and 1Hz (low-frequency data) for aggregated power, whereas reading for sub-metered individual appliances were recorded for every 3 or 4 seconds.

**Strengths:** The datasets combined both aggregated home power consumption and individual appliance power consumption which are very useful in training both supervised and unsupervised learning for load disaggregation. Moreover, the dataset contains data for 6 different houses, which could help to validate the applicability of any load disaggregation-based approaches.

**Weaknesses:** The datasets came with aggregated and sub-metered electricity consumption sampled with low frequency (1Hz) which reduces the length of signal features. The higher the sampling frequency, the more signal features are recorded by the smart meter.

**Overview:** Murray et al. [17] presented the REFIT (Personalised Retrofit Decision Support Tools for UK Homes Using Smart Home Technology) that combines the whole house

aggregated electrical loads and 9 individual appliances load readings for 2 years (Sept 2013-July2015)) duration with the load sampling rate of 8 seconds for each house in 20 houses.

**Strengths:** The data contain house and appliances level energy information across 20 houses, recorded for 2 years. This data can help in comparing the energy consumption behavior across different houses and seasons.

**Weaknesses:** The dataset comes with low sampling frequency which reduces the length of signal features. Moreover, having data with low sampling frequency limits some load disaggregation-based method in separating the appliance with complex signature features. The higher the sampling frequency, the more signature features could be extracted by the load disaggregation method.

**Overview:** Makonin et al. [18] presented the first version of the Almanac of minutely power datasets (AMPDs) to be used in load disaggregation study. The data contains real power (P) readings for 1-year (Ap1,2012-March 31,2013) with a one-minute (60 sec) sampling interval for 21 sub-metered appliances for one house in British Columbia, Canada.

**Strengths:** The data provide power consumption of 11 measured important home appliances (such as fridge and microwave) for a good sampling frequency (1 min) which makes it fit for load disaggregation problems. The datasets contain information about natural gas and water consumption data, and this makes it usable for an extended study in managing the home energy.

**Weaknesses:** The datasets contain recorded information for one house, and this does not help in evaluating the applicability of load disaggregation based where the model disaggregation ability could be tested with different house houses and datasets.

**Overview:** Makonin et al. [19] published the power consumption for 3 years (2012-2015), weather data from the nearest weather station, and one-year data from simulated solar systems. Providing solar data could be used in analyzing the cost associated with installing battery storage for DC solar production in communities

**Strengths:** Dataset provides energy information's for a long duration and contains weather information which is the key factor in load disaggregation and load forecasting problems. Providing solar power consumption along with weather data could help in solar energy forecasts especially for power systems with solar systems generation to limit the cost associated with storage batteries.

**Weaknesses:** The datasets come with records for one house that prohibits the transfer learning for load disaggregation in which model is trained using data from one house and applies to other houses to limit cost associated with installing sensors for each appliance to obtain the corresponding readings.

**Overview:** Kelly et al. [20] provided the UK domestic appliances -level electricity dataset (UK-DALE) that comprises the household data for both aggregated and sub-metered individual appliances from 4 households. The data were recorded for 4 years and contains voltage, current, and real power parameters with 16 KH (for Voltage and current) and 6 sec (for power) sampling frequency.

**Strengths:** The dataset provides a way to evaluate the applicability of any NILM based model because it comes with appliances power consumption across 4 different houses. It also contains features recorded with both high and low sampling frequency that help many load disaggregation-based approaches to discover the signal features of an individual appliance.

**Weaknesses:** The problem lies in the fact that the data are in different formats which makes it hard to compare results from developed load disaggregation approaches because researchers may use different file formats.

**Overview:** Makonin et al. [21] made available the second version of AMPDs datasets and this data globes electricity, water, and natural gas consumption for 2 years (April/2012 to March/2014) and gives 11 measurements of electricity based features recorded for a one-minute interval.

**Strengths:** The dataset provides a unique aspect because it presents data that contain power consumption of natural gas and water for a long duration of 2 years. Apart from other datasets that come with reading for active power, the dataset provides data with enough electricity-based features (11) which could contribute to discovering the signal of an individual appliance. The dataset was preprocessed, and this can contribute to the accuracy of load disaggregation-based approaches.

**Weaknesses:** It contains information for a single house, and this does not help in NILM model transfer learning and testing for load disaggregation problem. Training and testing load disaggregation model with appliances from a different house is important for model generation and adoption.

Table 2.1:Details of publicly available datasets for electric energy disaggregation

Author	Dataset &year	location	evaluation period	# House	appliances	Appliance Sampling rate	Total power sampling rate
Batra et al. [14]	2014 iAWE	INDIA	73 days	1	33 devices	1sec or 6 sec	1 sec
Candedo et al. [15]	Kaggle, 2016	Belgium	4.5 months	1	N/A	10 min	10min
Kolter et al. [16]	REDD, 2011	USA	3-19 days	6	24	3 sec or 4 second	1 Hz, 15KHz
Murray et al. [17]	REFIT, 2013	UK	2years	20	11	8 sec	8 sec
Kelly et al. [20]	UK-DALE 2014	UK	2.5 years	5	5-54 devices	6 sec	1-6sec and 16 kHz
Makonin [18]	AMPDS 2013	CANADA	1 year	1	19	1 min	1min
Makonin et al. [21]	AMPds2 2016	CANADA	2 years	1	21	1 min	1min
Makonin [19]	HUE, 2019	CANADA	3 years and 1 year for solar	9	Electrical energy	Electrical energy	1h

## 2.2. Review on Existing Load Forecasting and Disaggregation Approaches

The Literature review starts by reviewing the existing load forecasting approaches that learn from weather parameters from inside and outside of the house to predict the appliances' electricity consumption in the house. The section also covers the existing machine and deep learning approaches that separate the aggregate house power demand from the individual appliance's energy consumption.

### 2.2.1. Review on Load forecasting approaches

Machine learning techniques are used to solve the load forecasting task where prior energy data and weather data are used to train the model and forecast the future household power demand.

Different techniques have been used to forecast the appliances' power consumption.

**Overview:** Platon et al. [22] presented hourly load forecasting using artificial neural networks (ANN) and case-based reasoning (CBR) on the building's electricity consumption and weather data. To reduce the number of inputs parameters to the model, the authors used a statistical modeling technique called principal component analysis (PCA) which filters out the input parameters based on their significance to the house energy consumption. The hourly forecasting results indicated that for a 6 h duration, the ANN outperforms the CBR model with an error of 7.3% compared to 13% of CBR, but interestingly, both ANN and CBR forecasting error were found in the standard limit(30% for hourly prediction) for the data-driven model that is recommended by the American Society of Heating, refrigeration and air-conditioning engineers(ASHRAE) [23].

**Strengths:** The advantages of the study are seen in applying the principal component analysis (PCA) technique that helps in deciding which input parameters to be included in the model depending on the information contained in the parameters. The use of the PCA technique helped in achieving an interesting model forecasting performance by inputting 10 significant features out of 22. For example, using 10 inputs selected with PC techniques, the root mean square error was reduced to 13.37% compared to 14.32% obtained by training the model with 22 inputs. Utilizing input filtering techniques such as PCA could help in reducing the effect of noise and unnecessary inputs to the model's ability to accurately predict future energy consumption.

**Weaknesses:** No comparative analysis between data preprocessing based on PCA technique and other techniques such as load disaggregation. Also, the study used one evaluation metric (RMSE), which limits the performance generalization.

Overview: Candanedo et al. [15] compared the load forecasting ability of 4 models (multiple linear regression(MLR), support vector machine(SVM) with the radial kernel, random forest and gradient boosting machines (CBM) by inputting electricity consumption of appliances, weather (indoor and outdoor ) inputs and recorded lighting load that were recorded for 4.5 months duration. The results showed that CBM was able to better learn from the variance among input parameters compare to linear regression and vector support machines. In terms of RMSE, the CBM achieved 66.65 compared to 70.75 of SVM radial and 93.18 of MLR.

The features important analysis revealed that the weather-related inputs are key important for energy prediction. The atmospheric pressure was found to be the most effective in building energy prediction since it is dependent to wind speed and rainfall which increase the presence of home residents at home.

**Strengths:** The advantage of this approach is found in the application of the Boruta algorithm that helps in filtering and selecting the variable based on their effect on the model performance. The data filtering algorithm helps in reducing the complexity of the developed model.

**Weaknesses:** The study used weather-related information's obtained from a weather station located far from the house and this contributed to the low accuracy of load prediction.

**Overview:** Qiuyu et al. [24] presented a short-term load forecast that considered the impact of meteorological weather data to the house power consumption. During data preprocessing, the data were divided into time series load trends and weather-sensitive

(daily and weekly) parts and a support vector regression(SVR) approach was used to learn from the weather-sensitive load component whereas a Holt-Winters model was used to understand the variation of time series base component of the load. The one day ahead load forecasting results indicated that the forecast was increased by incorporating the weather and meteorological inputs into the SVR model. The mean percentage error (MPE)results for the time series Holt-Winters Model were high (2.85%) compared to 2.23 % of the SVR approach that considered the weather and time and weather categorical data (daily, holidays).

**Strength:** The study showed the benefits of adding weather information and time of energy use into short- term load forecasting. The modeling of the weather-sensitive load component demonstrated the crucial impact of weather conditions on the change in energy consumption in distributed systems.

**Weakness:** The model results were not compared with other machine learning algorithms for load forecast so that the performance comparison can be clarified. The model was trained using one summer term (4 months: May to September 2017) weather information's which limited exploring the impact of weather conditions on yearly energy consumption.

**Overview:** Dehalwal et al. [5] compared the load forecasting strength of bagged regression trees with that of artificial neural network (ANN). The two models were trained using the urban area load and weather forecasts from the Australian Energy Market Operator (AEMO) and Meteorology of the Sydney region from 2006 to 2009. The results showed that the Bagged regression trees achieved low mean percentage error (MAPE:1.54%) compared to 1.90% of ANN

**Strength:** The study used a long duration dataset (3 years) which is a key in understanding the nonlinearity in the input information as well as the seasonal variation of power demand to the weather changes. Feature engineering based on chronological data and meteorological inputs helped in categorizing the input data and understanding the impact of weather on the power demand.

**Weakness:** The study used the prior aggregated house energy consumption which does not represent the complete behavior of home residents. Using appliances level data can increase the training window data and enhance the accuracy of the forecasting.

**Overview:** Ma et al. [25] developed a long short-term memory (LSTM) based load forecasting model that used iForest for feature engineering and cleaning of the prior load data. The results indicated that the preprocessing with iForest helped the LSTM to achieve a good load forecast with a low mean percentage error (MAPE: 0.92%) compared to 1.15% of a single LSTM.

**Strength:** The study demonstrated how the isolated forest can help in detecting and filtering the uncorrelated data information's. The use of isolated forest to detect and remove uncorrelated features have resulted in increasing the LSTM prediction performance (MAPE:0.92%) compare to 1.15% of a single LSTM. The study used a long duration (Jan 2016 to Jan 2017) dataset with 15 min sampling frequency and 35712 samples and predicted the load forecast for 7 days ahead.

**Weakness:** The study did not pay attention to the impact of meteorological weather conditions on energy consumption. This limited the model to learn from both prior energy consumption and outside climate weather conditions.

**Overview:** Ebrahim et al. [6] designed and implemented the load forecast based load disaggregation. They used the combinatorial optimization to extract the energy consumption from the aggregated house power demand and train the feed-forward neural network (FF-NN)-based load forecast with appliances data. With this data preprocessing techniques, the FF-NN performed well and reduced the one day ahead forecasting error to an extent of 0.0877 in terms of normalized root mean square error.

**Strength:** This study experimented with a key approach to addressing the variability of house power demand by including appliance level data in the forecasting models. Training the load forecasting model with data that contains the energy consumer's behavior has resulted in reducing the forecasting accuracy.

**Weakness:** The model did not include the outside weather information and those inputs are important to the house energy consumption. For example, when it is cold, the home occupants can turn on the heating systems and increase energy consumption. Therefore, it essential to include nearby weather information in the load forecasting model.

The literature on load forecast can be compared by input data and forecasting techniques: With regards to the input data, the existing load forecasting is trained with time series aggregated load. As input data have similar patterns, the model training is simple. The results of this kind of training are not accurate as they do not include all power-related variables such as weather information. Secondly, the forecasting model is trained with related variables such as weather data collected at the nearby weather station or inside the house, and the model learns from the relationship among variables to forecast the future power demand. Finally, the load forecasting models can be trained with data obtained by data extraction techniques such as the load disaggregation. Data extraction based on load

disaggregation aims to obtain the appliances level information from the historical aggregated load and use them to train the load forecasting model. Compare with the mode trained with time series total load, training the model with appliances level data can increase the performance of the load forecasting model since the model is trained with data that illustrate the energy client's behavior.

In terms of forecasting methods, the regression techniques that are constructed to learn from the power-related variables are the most common approaches. The regression-based techniques such as the random forest [15], ensemble learning, and support vector regression were used and achieved good forecasting accuracy for short term forecasts especially when trained on time series total power. The studies [5,24] proved that the regression-based machine learning outperforms other single artificial neural network algorithms for short-term load forecasts. For example, bagged regression trees achieved low load prediction error (MAPE:1.54%) compared to 1.90% of ANN [5]. Also, the hybrid method that combines the potential of two or more individual forecasting methods is reported to reduce the load forecasting compare to a single machine learning or deep learning method [26].

### **2.2.2. Load Disaggregation based Approaches**

The literature on load disaggregation techniques covers probabilistic methods, optimization methods, and deep learning methods. Both probabilistic and optimization techniques need the aggregated total load and appliances level data for model training whereas the deep learning methods use unsupervised learning with no labeled appliance data.

In practice, the load disaggregation can be taken as semi-supervised learning that combines both supervised and unsupervised learning approaches. Supervised learning is used for model training where the model is trained with both house level and appliance level data and the unsupervised is used for model testing which uses the aggregated data [12].

The literature for the recent load disaggregation techniques is presented in this section:

**Overview:** Parson et al. [27] showed how the individual appliances' signal can be iteratively separated from the total load using the hidden Markov model (HMM). In this approach, the HMM is trained with both house and appliances level data to obtain the general appliance models and the prior models of general appliances types are turned to specific appliances instances using signatures extracted from the total load. After testing the model with US-REDD datasets, the results show that the disaggregation performance of the HMM model built with no sub-metered appliances data is closer to that with sub-metered appliances data. For example, for a refrigerator in house 1 of the REDD dataset, the average disaggregation error for the HMM model with only aggregated training is 15% compared to 14% for the HMM model with sub-metered training.

**Strengths:** The proposed approach achieved a comparative disaggregation accuracy without the need for sub-metered individual appliances data in the model training and the adaption of this approach can reduce the cost associated with collecting electricity consumption of each appliance present in the house.

**Weaknesses:** The model testing has been done using a single dataset and lacks a comparative analysis of how the model could disaggregated appliances from a different house whose information was not present during the model training.

**Overview:** Harell et al. [28] presented a dilated causal convolutional neural network in which the future inputs depend on current and previous samples while the output from each convolutional layer became the input to the next layer. By training the model with four input parameters (current, active power, reactive power, and apparent power) from AMPDs2 dataset, the convolution neural network achieved estimation accuracy (95%) compared to 94% of sparse super-state Hidden Markov Models(SSHMM) that used a single input(current).

**Strengths:** The study showed the strength of a convolutional neural network for learning from different energy features and reduce the variance among them for achieving an improved accuracy in estimating house power demand. The study added more inputs in the estimated model, which has contributed to the increased model performance.

**Weaknesses:** The study estimated the aggregated load and does not provide enough information to users on which home appliance is consuming more or the participation of individual appliances in household energy consumption. The study uses electricity-related features from one house in Canada and this limits the applicability to other houses. Furthermore, the study did not pay attention to the weather conditions and time of use which are the important parameters in load predictions.

**Overview:** Kelly et al. [4] compared 3 deep neural network architectures for estimating the appliance by appliance electricity consumption from a single smart-meter aggregated load. After training the models with UK-DALE energy data, the testing results (both interpolation and extrapolation) showed that any of the 3 methods outperform other factorial hidden Markov models in load disaggregation. For example, the accuracy score

for separating the microwave power signal was 0.98 for LSTM, 0.99 for rectangles thinking, 0.99 for autoencoder, 0.91 for factorial hidden Markov model (FHMM).

**Strengths:** The model testing was done on data (UK DALE) with high frequency (6sec) which is useful to extract a huge number of features. The second advantage is the use of different evaluation metrics to test the model performance which helps in performance generalization. Furthermore, the study addressed the applicability issues, because it has been tested on different houses across UK-DALE datasets.

**Weaknesses:** The problem is that the data from small appliances such as phone chargers were not included in the model training which means that their contribution to the aggregated house power demand was not considered. In large houses, those appliances could constitute a certain portion of the entire house power demand. In addition to this, the testing has been done on houses present in the same dataset (UK DALE), and this does not solve the scalability problem where a model can be applied on different houses in the same country but does not work in other countries because for example appliance models and specifications in the United Kingdom are different to specifications in the United States.

**Overview:** Rafiq et al. [29] presented the LSTM and gated recurrent unit (GRU), deep recurrent neural network model, to separate the individual appliance power consumption from the aggregated load. The testing was done using unseen building data from the UK DALE dataset. The regularized LSTM model achieved high performance in real-time energy disaggregation of home appliances compared to the GRU network. This is originated in the ability of LSTM to maintain its internal cell compare to the GRU network that exposes its entire memory. Compared to the results of other LSTM architecture presented in the literature [4], the results of two regularized (i.e. introduction of dropout)

models were found to be in the same range. In terms of the F1 score, the signal of the microwave was disaggregated with 0.98 scores for LSTM, 0.981 for GRU and 0.98 of LSTM reported in [4] using the same datasets UK-DALE.

**Strengths:** The study demonstrated the importance of utilizing the regularization techniques for removing the hidden units in time of model training to boost the disaggregation score. Also, the model succeeded with the disaggregation of both appliances with simple and complex signature features such as a microwave. The model testing was done on different houses in UK DALE which implies the applicability of the proposed model in real-time learning.

**Weaknesses:** The models were tested on separating signature features of appliances in the UK only and lack a comparative analysis to evaluate its applicability across appliances in other countries. This means that this model may work well with disaggregating appliances in the UK but can not work in disaggregating appliances from other regions or datasets such as the U.S.

**Overview:** Kim et al. [2] proposed the deep learning architecture which separated the input power signal using a reflection rate that extracts the appliance signal from another house power signal. The study presented an energy disaggregation LSTM based deep learning. The US-REDD and UK DALE datasets have been used to test the model. The results show that the proposed LSTM model outperforms FHMM in disaggregating signal features of appliances. For example, in terms of the F1 score, the LSTM network was able to disaggregate an air conditioner present in house 1 of the UK-DALE dataset, with 0.764 scores compare to 0.30 score of FHMM.

**Strengths:** The proposed model addressed the problem of disaggregating appliances features from low sample aggregated house data. The LSTM architecture was able to separate the appliances with multiple states and appliances with the same power consumption from the aggregate power consumption. Also, the model was able to sustain the performance on different houses from different regions or countries present in two different datasets (UK DALE & US-REDD) where appliances are with different models and specifications. The study solved the issues of applicability and scalability to achieve a complete model transfer learning.

**Weaknesses:** The study is limited in disaggregating the individual appliances from the total load and is with no load forecasts to predict the aggregated appliances' electricity consumption-based weather data which is also important in addressing issues of home energy management.

**Overview:** Bhotto et al. [30] presented aided linear integer programming (ALIP) model for load disaggregation which is an improvement to a conventional linear integer programming (IP). As opposed to the IP model that is based on waveform signature, more constraints such as median filtering and improved diagrams were added to the IP model to obtain the AIP model that performs well on low sampling data. The model performance testing has been performed on REDD and AMPDs datasets and the results show that the refined ALIP model outperforms the IP model. For example, the accuracy for a disaggregating load of refrigerator present in house 1 of the REDD dataset is 0.96 for ALIP compared to 0.92 of the IP model.

**Strengths:** The approach is tested on two different datasets from different countries: REDD of USA, and AMPDs of Canada which answer the applicability problem. In addition

to this, the model worked well with low frequency (3 sec) REDD datasets as well as 1 min of AMPDs dataset.

**Weaknesses:** The study lacks correlation analysis so that it can clarify the impact of each input feature on the house energy consumption. Also, the impact of weather data to the aggregated appliances consumption is not considered in the presented study.

**Overview:** Martin et al. [31] presented an industrial machine dataset and utilized this dataset to compare FHMM with a dilated causal convolution network in disaggregating energy consumption of six industrial machines from a single smart meter located in factory in Brazil. The results revealed that for the same appliances, the dilated causal convolution model was able to reduce disaggregated error compared to FHMM because, in dilated causal convolution model, the information flows in chain framework, from one layer to another layer until the output layer. The results show that for Milling machine1, the normalized signal aggregated error is 0.39 for FHMM and  $0.09 \pm 0.04$  for dilated causal convolution network.

**Strengths:** As opposed to other available models that work with residential or commercial datasets from developed countries, this study showed the possibility of implementing load disaggregation in developing countries such as Brazil. This is an interesting step in NILM adoption in the world.

**Weaknesses:** The drawback of the study was that it is tested on data obtained from a single factor which limits the model generalization for industrial application.

**Overview:** Silva et al. [32] demonstrated a way to pre-process data using an advanced filter pipeline and disaggregate aggregated load data signal using a probabilistic knapsack

algorithm, a partition map. The approach provided a way to label appliances and determine the electricity consumption of appliances from aggregated home electricity consumption and the model was tested on homes present in different regions and countries. The results indicated that the proposed method could sense the relatively complex signals of appliances and these signals account for 93.7% of the total aggregated energy consumption.

**Strengths:** The advantages of the study lies in the fact that the authors presented universal NILM composed with unsupervised and transferable learning in which the model learned from the house data and separate the electricity consumption of appliances located in the different house without training of prior appliances information in each house.

**Weaknesses:** The problem of the model is that it has been tested using a single dataset and could face the applicability problem because it could be able to disaggregating appliance located in one home in Canada but could not work in other countries such as the UK where appliances are with different model and specification.

**Overview** Mauch et al. [33] demonstrated a simple way to separate the power signal of a dominant appliance from the total power signal of the same time using the LSTM network. The approach considered the measured power signal of the targeted appliances along with the total power signal of the same period to train the LSTM network. After testing the trained model on house 2 of the REDD dataset, the model achieved the disaggregation F1 score of 0.66 for microwave.

**Strengths:** The model was able to disaggregate both the ON/OFF (e.g. microwave) and multi-state types of appliances such as dishwashers. This is an important step as hidden Markov based models fail to disaggregate multi-state types of appliances. Also, the study

solved the problem of difference in sampling frequency of the aggregated signal and sub-metered appliances by creating the synthetic data obtained by summing all sub-metered meters and this limited the noise because only appliances that consume more power were included in the model training.

**Weaknesses:** The study considered real power as the only input parameters and this could have contributed to the disaggregation performance. In addition to this, the LSTM model was tested on a single dataset which could limit its use to disaggregate houses located in different datasets or countries.

**Overview:** Murray et al. [34] presented CNN and GRU network architectures that classify state and estimate the appliances' power consumption independently on whether the house data were present in the training process. The models were trained on houses present in the REFIT dataset and tested on unseen houses present in both REFIT recorded UK and REDD recorded in the US. The results showed that CNN was able to disaggregate a refrigerator located in unseen REFIT house 2 at an extend of 76.9% compared to 76% disaggregation accuracy obtained by testing the CNN model on unseen US-REDD house 2. The drop-in model performance between UK-REFIT and US-REDD is originated from the difference in model and specifications of appliances in the UK house and the US house. Compared with the literature, the proposed model architectures show similar F1 score performance for microwave (0.95) and better score (1) for fridge compare to 0.94 of [34] using the same datasets REDD and tested on house 2.

**Strengths:** The study addressed the transferability of learning across houses located in different regions or countries by testing the model with two different datasets and this model applicability could help in reducing the cost of installing IoT sensors in each house

to extract the power consumption of individual appliances. Further, this could be a solution to electricity consumers located in developing countries where the ability to buy sensors is very low.

**Weaknesses:** The study did not evaluate the impact of meteorological weather conditions to house power consumption. Combining the benefits of having appliances signal and weather conditions could lead to an accurate load forecast in the household.

**Overview:** Mauch2 et al. [35] presented a way to extract a single load from the aggregated load without the prior knowledge of all loads present in the aggregated load. These have been possible by using two different observations where the single load is modeled using Gaussian observation whereas deep neural networks are used to model the aggregated signal.

**Strengths:** The study addressed the problem of the HMM model to disaggregate the total load without the need to include all appliances level data in the model training. The study showed the importance of constructing training that combines HMM with deep learning can make HMM applicable to large buildings and achieve good accuracy even if the number of appliances is increased in input data. In addition to this, instead of using the total smart meter load in the model training, the study used the input data obtained by summing up the power demand for all the targeted appliances and this limited the noise because the power of less important appliances was not included in the mode training. The study achieved small load disaggregation error for low-frequency REDD dataset.

**Weaknesses:** The model was only tested on a single dataset REDD and no comparative analysis that illustrate its effectiveness in disaggregating appliances from different datasets

or countries. The study did pay attention to outside meteorological weather conditions which cause the change in house energy consumption.

**Overview:** Humala et al. [36] presented a universal energy disaggregation technique that separated appliance energy consumption from aggregated power consumption independently to whether the targeted appliances information was present in model training. In this framework, the combinatorial optimization(CO) model is trained using few houses with both appliance level data and aggregated data, and the learning from those data is transferred and be used to test other houses with only aggregated power consumption without any training of the same dataset. The results show that the average RMSE obtained in disaggregating most appliances in both REDD and UKDALE is 12% and for microwave in house1 of UK DALE dataset (RMSE=0.16%). The CO energy disaggregation approach outperforms the reference model with an F1-score of 0.57 compared to 0.55 of the factorial hidden Markov model[37].

**Strengths:** The advantages of this universal based combinatorial optimization model are that they could be used to disaggregate other houses whose appliances information was not used in the model training which then resolve the model applicability. The fact that the model achieved a high load disaggregation performance on two different datasets from different countries, demonstrates the applicability ability which means that the model does not need more hand -turning to be applied across different households.

**Weaknesses:** The drawback of the model is that it did consider the impact of outside weather conditions to the house energy consumption. Moreover, CO achieved a low disaggregation score for complex appliances (e.g. washing machines) and appliances with power consumption similar to many other appliances.

**Overview:** Makonin et al. [38] presented a sparse NILM algorithm that can disaggregate multi-state appliances load in real-time using low sampling rates. The sparse load disaggregation method is constructed based on super-state model HMM and Viterbi algorithm to find dependencies between total load and appliances load to separate the appliances with multi states or complex signature. After model testing on house1 of the REDD dataset, results show that the super-state mode HMM model achieved more than 99.3% accuracy compared to 82.1% of Johnson 2013[39].

**Strengths:** The presented approach showed how the HMM model which was ignored by other researchers could achieve a comparative load disaggregation accuracy if the sparsity in matrix storage is processed using super state HMM. The improved HMM achieved high accuracy with low sampling frequency for load state classification.

**Weaknesses:** The disadvantage of the model is that it requires sub-metered appliances information which is not available as IoT sensors are not installed in all households across the world. Also, the model was tested with a single dataset and this limits its applicability to other households in the transfer learning process.

Table 2.2: Comparison of load disaggregation-based Approaches.

( P: active power; V: Voltage; I:Current; Q: Reactive power; S: Apparent power; HMM: Hidden Markov Model; FF-NN: Feed forward neural network; CNN: Convolution neural network; LSTM: Long short term memory; DAE: Denoising autoencoder; GRU: Gated recurrent Unit; AIP: Aided linear integer programming)

Study	Publication year	Used Datasets	Input features	Output	Training approach	Type of NILM approach	Tested across different datasets: Applicability
Parson et al. [27]	2011	REDD	P	P	HMM	Probabilistic	No

Mauch et al. [33]	2016	REDD	P	P	FF-NN and HMM	Probabilistic and Machine Learning	No
Ebrahim et al. [6]	2018	UK-DALE	P	P	CO	Optimization	No
Humala et al. [36]	2018	UK-DALE and REDD	P	P	CO	Optimization	Yes: Applicability & Scalability
Harell et al. [28]	2019	AMPDs2	I, P, Q, S	P	WaveNet CNN	Deep learning	No
Kelly et al. [4]	2015	UK-DALE	P	P	LSTM, DAE, CNN Rectangle	Deep learning	Yes: Applicability
Rafiq et al. [29]	2018	UK-DALE	P	P	LSTM, GRU	Deep learning	Yes: Applicability
Kim et al. [2]	2017	UK-DALE and REDD	V, I, P	On/off	LSTM	Deep learning	Yes: Applicability & Scalability
Bhotto et al. [30]	2017	REDD & AMPDs2	P	P	AIP	Optimization	No
Murray [34]	2018	REDD, REFIT, UK DALE	P	P, on/off	GRU, CNN	Deep learning	Yes: Applicability & Scalability

The existing load disaggregation-based approaches fall in 2 categories:

The first category includes the optimization [6,36] and probabilistic [27,33] based approaches that require the labeled appliances data in the model training. The optimization method works by minimizing the error between the aggregated total power and the sum of individual appliances energy consumption whereas the probabilistic methods break the aggregated load into several appliances' load. These two methods are accurate and have

been considered as the NILM benchmark algorithms to evaluate any other load disaggregation approach.

The second category includes unsupervised or semi-supervised learning-based approaches that are trained without or a small number of appliances data to extract the appliances signatures from the aggregated total power. The deep learning-based architectures [4,28,34] have been reported to succeed in disaggregating the appliances with complex or multi-state signal features such as washing machines compared to probabilistic and optimization algorithm. The study also proved the ability of deep learning to separate the appliance features from the total load for an unseen house which leads to the applicability and scalability of the developed NILM based approach.

## CHAPTER 3: DATASET DESCRIPTION

This chapter provides the information and visualization of three publicly available datasets. Section 3.1 gives a visualization of the Gokagglers dataset for the load forecast. Section 3.2. presents a data visualization of the reference energy disaggregation dataset (REDD). Section 3.3 gives the visualization of the almanac minutely power dataset (AMPDs) for load disaggregation.

### 3. 1 Gokagglers dataset for aggregated appliances energy prediction

This subsection presents the information and visualization of the Gokagglers dataset that are applicable for testing load forecast models.

#### 3.1.1 Information in Gokagglers dataset

The Gokagglers dataset contains energy information that was collected with 10 minutes sampling rate from the smart meter installed in a single house in Belgium. The energy data was recorded for a 4.5 months duration and is accompanied by the weather information measured from inside the house and the nearby Chievres Airport weather station.[15].

The table below shows the input features for the load forecasting-based approaches.

Table 3.1: Features information in Gokagglers dataset. Taken from [15]

Name of Feature	Description and unit	datatype
Date	date time year-month-day hour: minute: second	Time Series, timestamp
Appliances	Aggregated appliances energy use in Wh	Timeseries, double
light	Energy consumption of lighting load inside the house and is recorded in watt hour (Wh)	Timeseries, double
Temperature inside	Temperature measured inside the house kitchen and is recorded in Celsius	Timeseries, double

Humidity inside	Humidity measured inside the house kitchen and is recorded in %	Timeseries, double
Temperature out	Temperature measured at Chievres weather station and is recorded in Celsius	Timeseries, double
Pressure out	Pressure measured at Chievres weather station and is recorded in mm Hg	Timeseries, double
Humidity out	Humidity measured at Chievres weather station and is recorded in %	Timeseries, double
Wind speed	Wind speed measured at Chievres weather station and is recorded in m/s	Timeseries, double

### 3.1.2 Data visualization of Gokagglers dataset

As shown in figure 3.1, the aggregated appliances energy consumption varies depending on the time and day of use; more appliances energy consumption is observed during the weekend and days off, while the low power consumption is observed in nighttime and weekdays as more appliances are turned off or in standby mode. As depicted in Fig 3.1 and Fig 3.2, the appliances mean power consumption is high on Mondays, Saturdays, Fridays, and Sundays. Figure 3.2 and figure 3.3 demonstrate that appliances consume more electricity on weekends than on weekdays. Moreover, the higher energy is consumed in wintertime (January) than in summertime (May).

Based on the hourly mean energy consumption shown in figure 3.4, it is clear that between 10 pm to 7 am, the consumption of the appliance is below 80 Wh, which means that most appliances are turned off or standby; from 7 am to 1 pm appliances energy consumption is increasing because most appliances including microwaves, kettle, oven are in use, but the appliances' energy reduces to 110W after lunch. The maximum appliances energy consumption is increased again (135 W to 180 Wh) in the afternoons as all family members are at home to use most of the appliances.

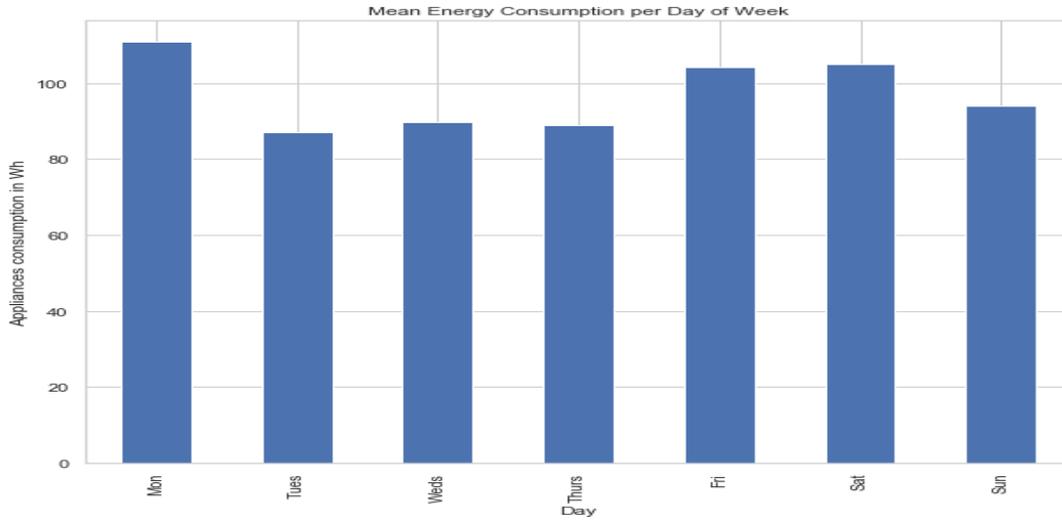


Figure 3.1: Daily appliances power consumption

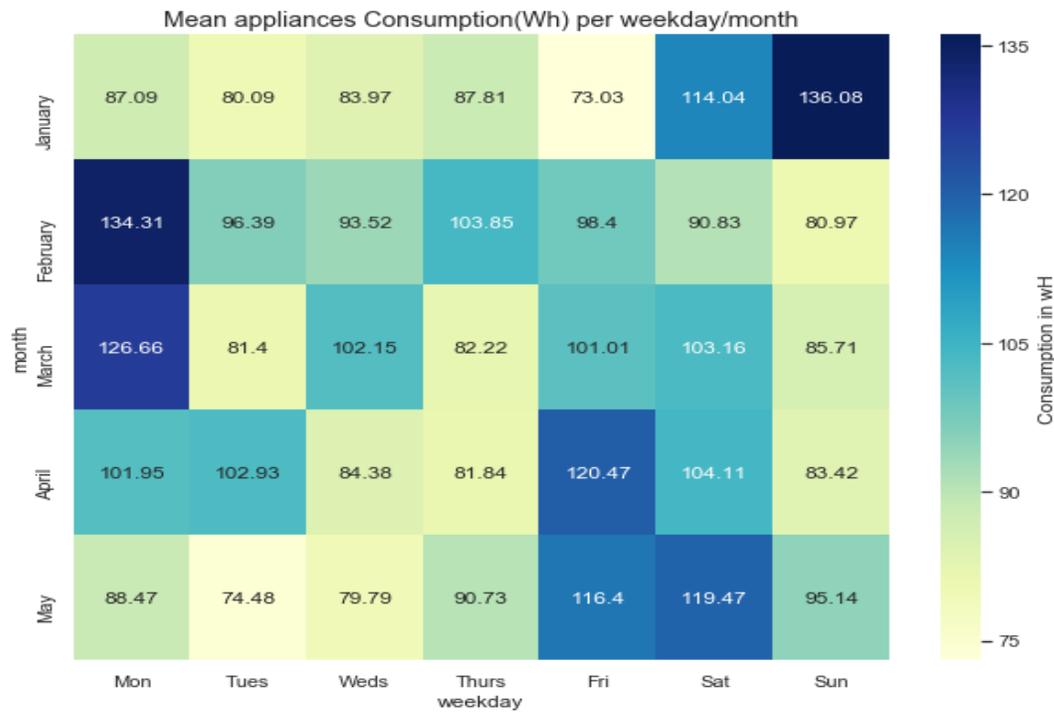


Figure 3.2: Weekly mean appliances power consumption per week

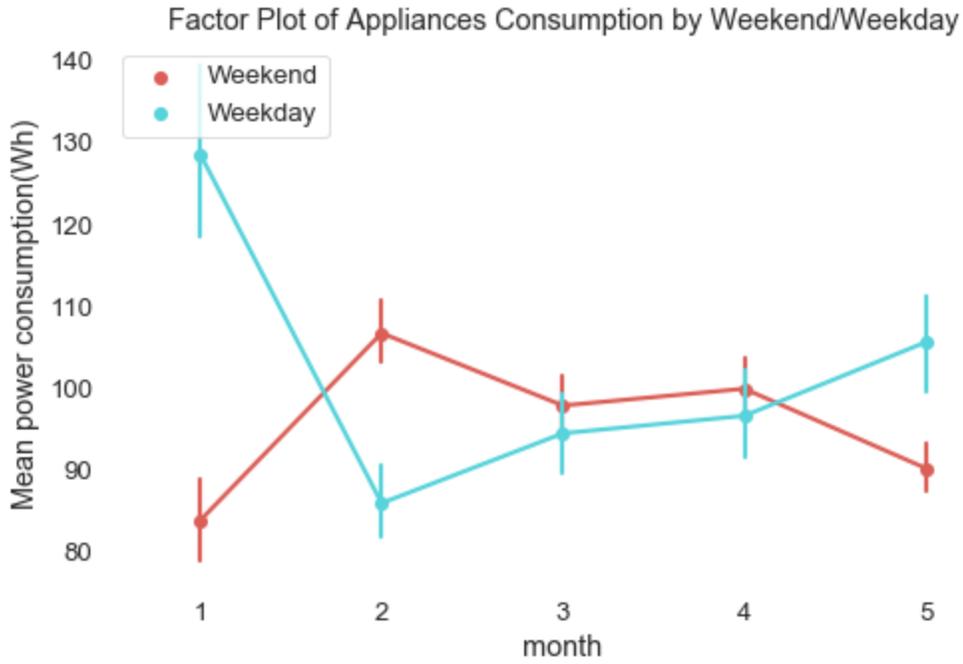


Figure 3.3: Appliances' weekly mean power consumption every month

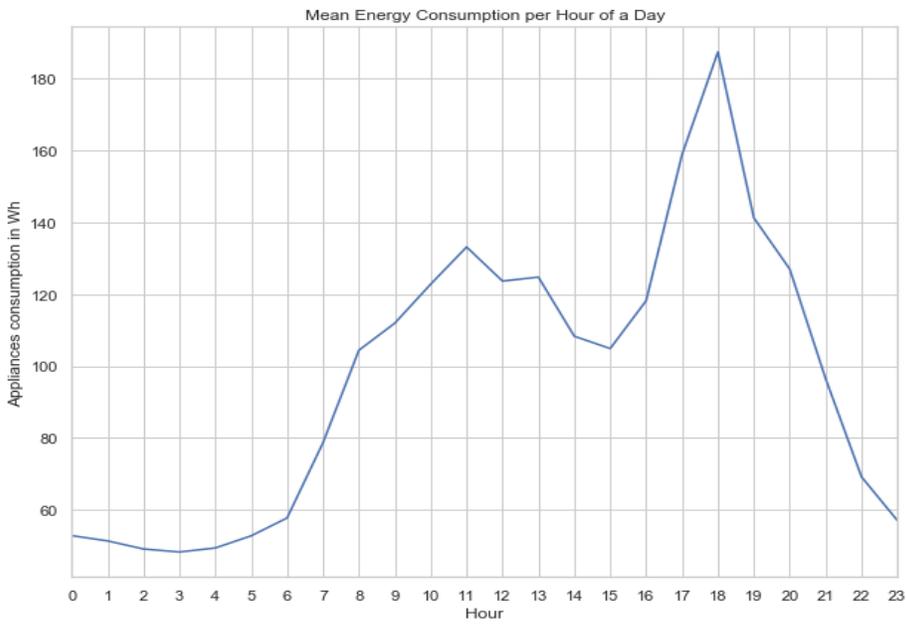


Figure 3.4: Appliances hourly power consumption

Figure 3.5 indicates the appliances' energy consumption is positively correlated with the lighting load, and weather parameters.

The appliances' energy consumption is positively highly correlated (0.2) with lighting and shows a very slight positive 0.099 and 0.055 correlation with outside temperature and inside temperature, respectively. Moreover, there exists a negative correlation between the appliance's electricity consumption and outside humidity (-0.15), and outside pressure (-0.035).

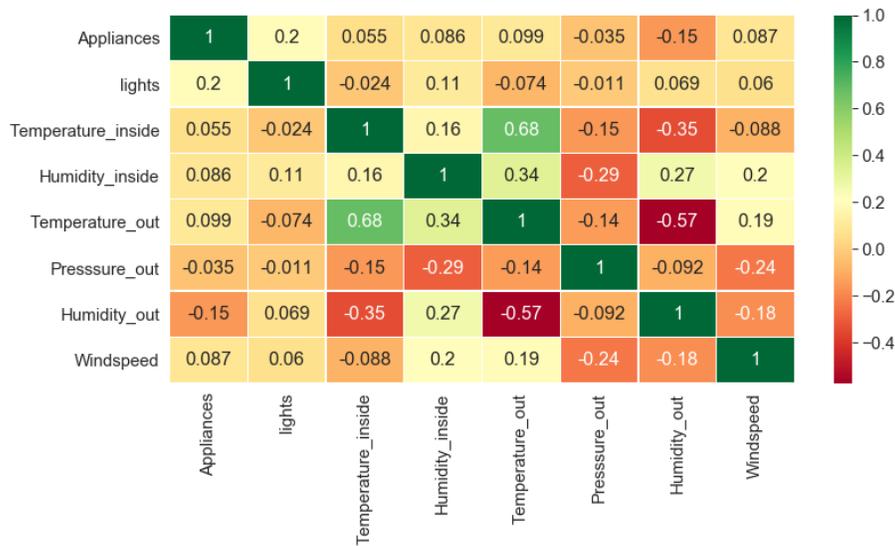


Figure 3.5: Correlation between appliances consumption and in-door and outdoor features in Gokagglers dataset.

### 3.2. Reference Energy Disaggregation Dataset (REDD)

This subsection covers the information and visualization of the REDD dataset that contains both house and appliance level energy data.

#### 3.2.1 Information in REDD dataset

This data contains information about electricity consumption of aggregated(mains) and appliance level data that were collected from six houses in Massachusetts, United States of America for a duration of 19days. The REDD dataset is the first and the most popular dataset used by the NILM research community since it was made available to the public in

2011 by Kolter& Johnson. The dataset contains low frequency (1Hz for total power and 3 or 4 seconds for appliances level data) power data and high frequency (15 Hz for current and voltage) waveform [16].

Table 3.2 shows the sub-metered appliance information used in the developed load disaggregation approach.

Table 3.2: Features information in REDD dataset. Taken from [16]

Number of Houses	Type of appliances in the house	Datatype
6	Fridge, microwave, dish washer, sockets, light, washer dryer, furnace, air conditioner, disposal, kitchen outlets, stove, waster disposal unit, etc.	Timeseries, double

### 3.2.2 Visualization of REDD dataset

The electricity consumption for aggregated (site meter) and sub-metered individual appliances present in house2 of the REDD dataset is shown in figure 3.6. As shown in figure 3.7, more electricity consumption is observed for the fridge, washer dryer, and lighting, whereas less electricity consumption is obtained for the waste disposal unit and electric stove. This originated from the fact that the fridge is always on an appliance as opposed to the electric stove that falls in ON/ OFF state type of appliances. The one-day electricity consumption of the ON/OFF state type of appliance (microwave) is shown in figure 3.8.

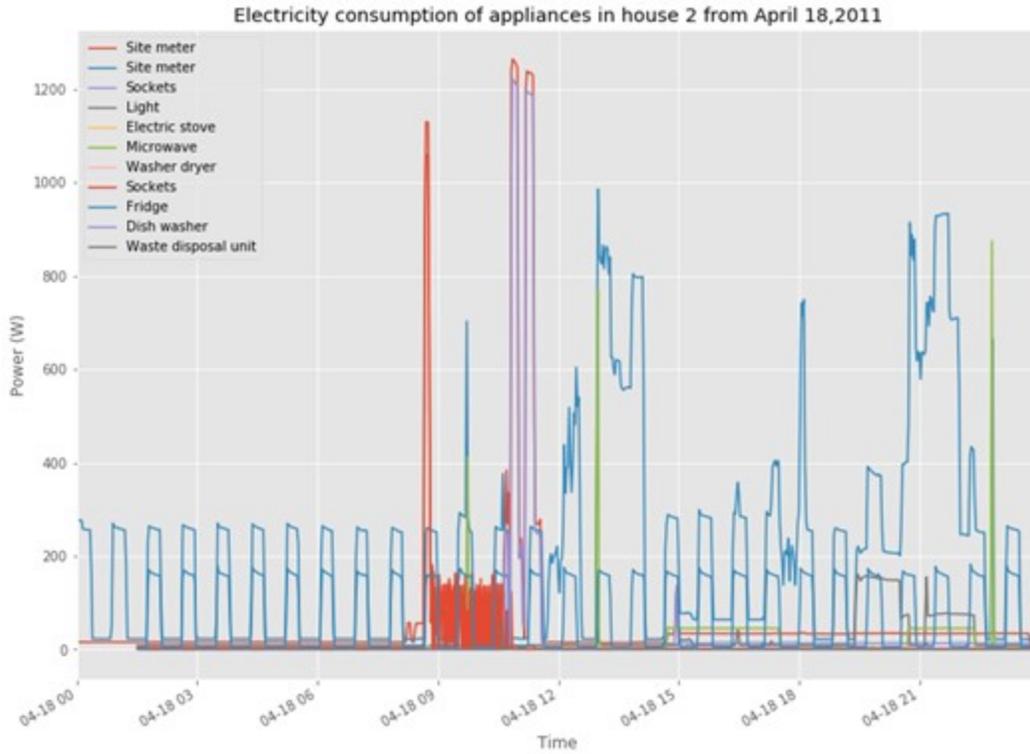


Figure 3.6: Aggregated and appliances energy consumption from house2 of REDD dataset.

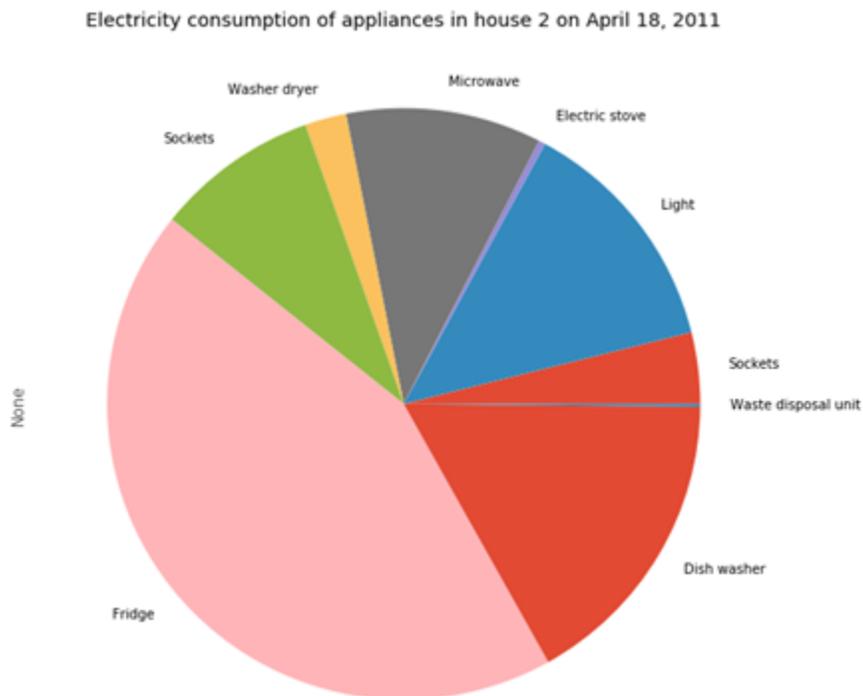


Figure 3.7: Appliances energy consumption from house2 of REDD dataset.

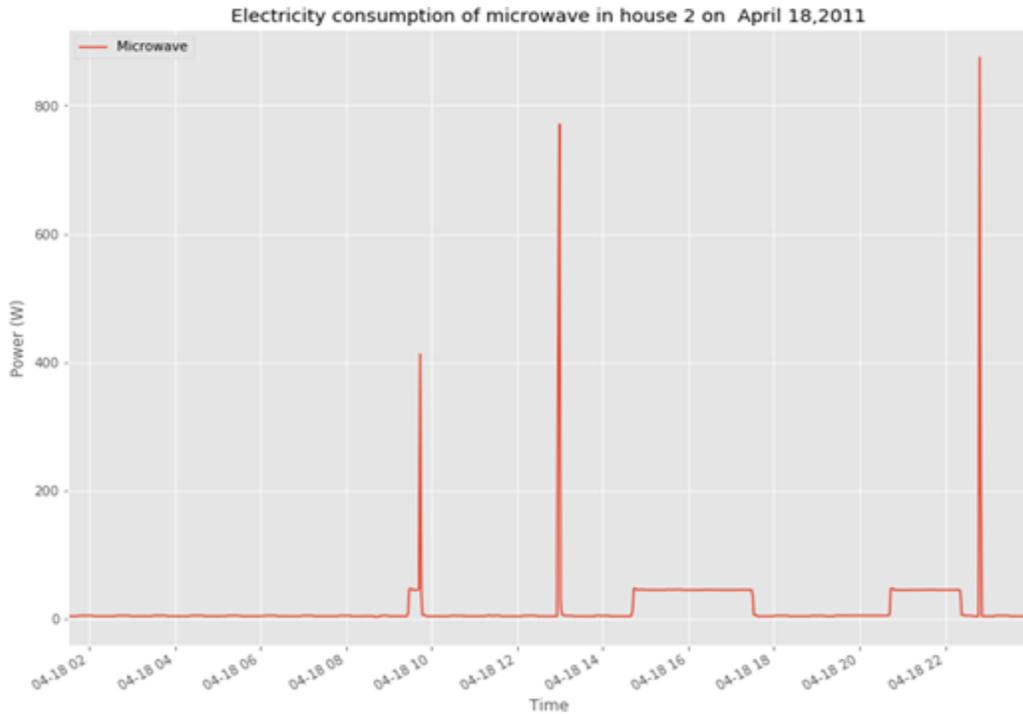


Figure 3.8: Microwave energy consumption from house2 of REDD dataset.

### 3.3. Visualization of Almanac of minutely power datasets (AMPDs)

The AMPDs [21] contains both the house and appliances level energy consumption of one house in Canada, recorded every minute for 2 years. It also has the hourly weather data information collected from the environment Canada's YVR weather station (Vancouver international airport).

The hourly total home energy consumption for both working and non-working days is shown in figure 3.9. As shown figure 3.9 and figure 3.10, The house consumes more energy in the morning (i.e. between 12 am to 8 am) time and in the later hours of the day. The less home energy consumption is observed in the middle day time (i.e. between 8 am and 1 pm). This can be explained with the fact that more appliances are likely to be in ON state

when people are back from the work in the evening time than in the work period. Moreover, as shown in figure 3.10, there is not much difference in the daily energy consumption.

The energy consumption of 7 important appliances is illustrated in figure 3.10, and it has been observed that heat pump and air furnace and thermostat cover half of the total home energy consumption.

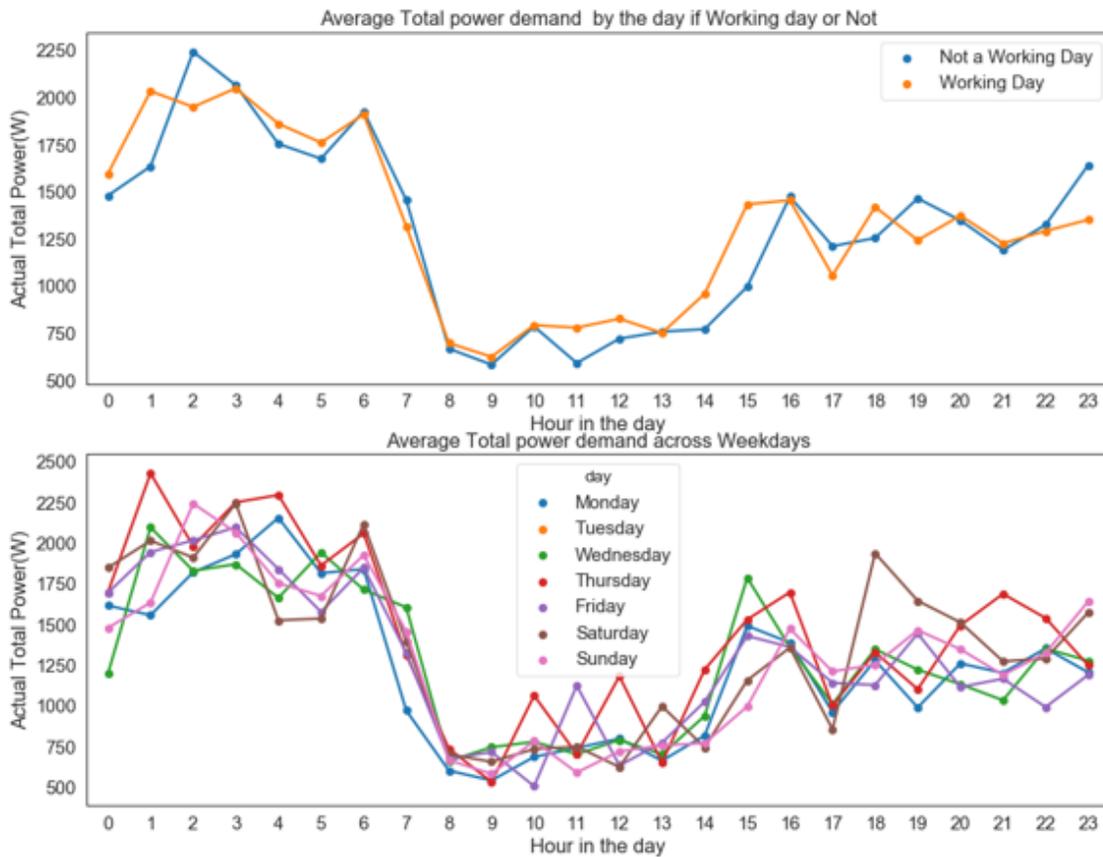


Figure 3.9: Hourly aggregated house power demand in AMPs

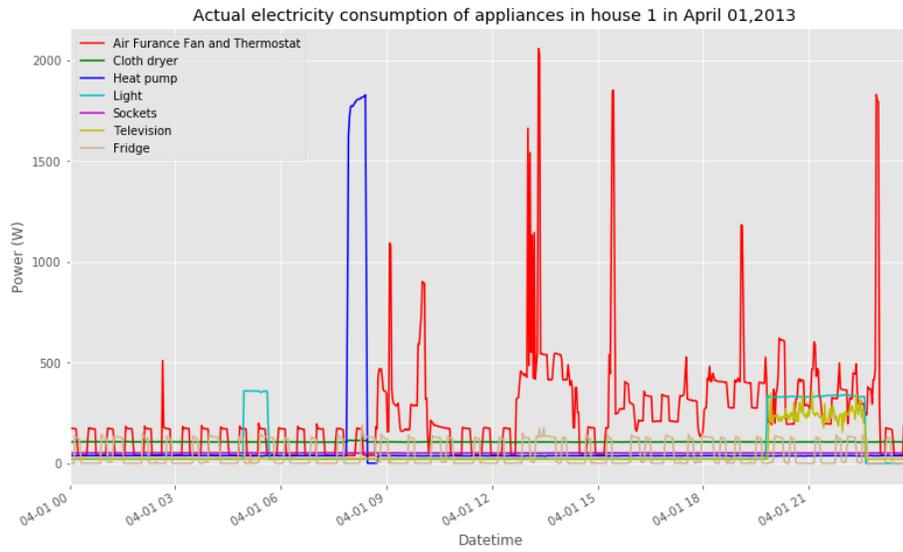


Figure 3.10: Energy consumption of 7 important appliances in AMPds

## CHAPTER 4: GAP ANALYSIS, RESEARCH QUESTION, AND PROBLEM STATEMENT

This chapter presents the gap in the literature, research questions, and the contributions of this thesis to the existing load disaggregation and forecast. Section 4.1 gives the gap in datasets used for load disaggregation. The gap in techniques for load disaggregation and forecast is presented in section 4.2. The research questions for this study are presented in section 4.3. Section 4.4 describes the problem statement. The contributions of this thesis to the load disaggregation and forecast are presented in section 4.5.

### **4.1. Gap in Existing Load Disaggregation based Datasets**

The gap in the existing dataset for load disaggregation is seen in the geographical location of data collection; several houses in the dataset (i.e. amount of data) and data sampling frequency.

With regards to the geographical location, most of the publicly available dataset are from western countries with the many datasets collected in the USA [16], Canada [21,19], and European countries such as the UK [17]. Few datasets were found to be collected in developing countries such as India [14]. The availability of NILM datasets in different countries with appliances with the different specification is interesting for applicability evaluation of any developed load disaggregation-based techniques.

In terms of the amount of data (i.e. number of appliances) in the NILM datasets, some datasets [16,17] contain the energy records of at least two houses which can be used in evaluating the applicability of NILM based approach. In addition to this, the dataset [21]

comes with the weather information of the house nearby weather station which is an additional key in evaluating the load forecast-based approach.

Depending on the data sampling frequency, most of the existing dataset [14,16,17,20] contains energy records that were sampled with low frequency (i.e. below 10 seconds). The sampling frequency is a key important for extracting the features of appliances. The higher the sampling rate, the more detailed features the smarter meter can detect from the target home appliance.

Considering the above comparison, the dataset that contains records with high sampling and could cover the long evaluation period could be worthy enough to train and evaluate the load disaggregation-based approach. Moreover, adding weather data can contribute to fully tackle of energy management problem. Therefore, our proposed solution has been tested on a dataset with a comparative sampling frequency (1 min) that contains both energy and weather information.

#### **4.2. Gap in existing Load Forecasting and Disaggregation based Approaches**

Different existing load forecasting-based approaches are trained on either total home energy consumption or data from weather stations but are not trained with the home appliances 'level data. Including the appliances' -level information such as the energy consumed by lighting, heat pump, and a fridge can help the forecasting model to learn from the energy clients' behavior and reduce the forecasting error.

The cheapest way of obtaining the appliances 'level data is by load disaggregation in which different computational techniques are used to separate the house total power with the individual appliance energy consumption. Both supervised and unsupervised learning have

been applied for load disaggregation. The supervised learning-based techniques need the appliances level data for model training while the unsupervised or semi-supervised techniques are trained without or small appliances level data. The supervised learning-based methods such as the probabilistic and combinatorial optimization can succeed with classifying and disaggregating the signal features of the ON/OFF type of appliances such as microwave whereas the semi-supervised learning-based methods are found to be more useful for disaggregating the appliances with complex signature features such as the multi-state(i.e. washing machine) type of appliance. Obtaining the comparative accuracy for the unsupervised learning techniques requires designing a sophisticated architecture that can learn from the aggregated energy signal and extract the signal of individual appliances without labeled appliances data.

Regarding the dataset information, active power is the most used energy signal for training and testing the load disaggregation. The other signal such as the current and voltage could be also used to better represent the electric load. Moreover, instead of using the raw energy signal, different computational techniques could be used to obtain averaging data such as the mean power. Data visualization could also be performed to illustrate the appliance's current and voltage fluctuations.

Despite low performance for disaggregating appliances with complex features, probabilistic and combinatorial optimization-based approaches can succeed for most classical appliances that are characterized by an ON/OFF state of operation. The appliances level data contains repetition or similar patterns as the appliances may be found in standby energy consumption. Therefore, combinatorial optimization can be modified to address the

issues of similarity among energy patterns, filter out input samples, and improve the accuracy of load disaggregation.

In terms of load disaggregation approaches, literature papers have presented different approaches that provided different disaggregation accuracy and it is hard to compare them as they were tested on a different dataset. Therefore, there is a need to evaluate the effectiveness of machine learning and deep learning for disaggregating signal features of appliances using the same dataset.

In terms of load forecast based approaches, there is a need to provide a comparative analysis between the impact of appliances energy use and weather conditions on the house energy consumption. Moreover, it is required to evaluate how stacking modeling that combines different learners can relate the variance between appliances energy and weather parameters to predict the future house energy consumption.

### **4.3. Research Questions**

The objective of the proposed research is to identify when and how home appliances consume the energy signal and show statistics of the appliance energy use. This study is intended to develop a load forecasting that combined the benefits of load disaggregation and weather information's to address the variability of house power demand and reduce the short-term load forecast.

This research is meant to answer the following research questions:

1. When do the home appliances consume electricity and how appliances' energy consumption is related to human activities?

2. To what extent the appliance power consumption could be separated from aggregated house power demand?
3. Does adding the appliances level data in the model training increase the accuracy of the load forecasting model?
4. To which extent does stacking modeling could reduce the short-term forecast error compared to a single learner?

#### **4.4. Problem Statement**

In this thesis, the load disaggregation task is considered as the optimization that is aimed to minimize the mismatch between the aggregated house power demand and the sum of individual appliance energy consumption. The study targets the appliances with a significant contribution to the total house power demand. Considering that the aggregated house power demand contains the energy signal of all active appliances in the house, finding the energy consumption of important appliances in the house requires to minimize the noise component from other non-important appliances.

Let assume that  $\hat{y}(t)$  is the aggregated house power demand,  $y^{(i)}(t)$  shows the energy signal of an individual appliance  $i$ ,  $T$  is the forecasting period, and  $e(t)$  is a noise term. This noise term represents the contribution of other appliances with less energy consumption present in the house such as ground fault circuit (GFI)outlet. Therefore, the objective function is to minimize the error or noise term between the aggregated house power and the sum of individual important appliances energy consumption [40].

Minimize:

$$e(t) = |\hat{y}(t) - \sum_{i=1}^T y^{(i)}(t)| \quad (4.1)$$

Assuming that if input pattern  $y_i$  from the prior house and appliance level data is similar to the other patterns  $y_j$  in the same estimating period, then the estimated output of  $y_i$  is similar to that of  $y_j$ . As the time series may contain repetition and sample of similar shape, the objective is to filter out prior energy consumption of appliances and reduce the complexity of forecasting task.

By considering that: (1) An individual appliance (i.e. Heat pump) can only be found in one house and (2) the aggregated house power demand is equal (i.e. in the denoised scenario) or less than the sum of individual appliance's energy consumption.

The appliances' energy consumption represents human activities and behaviors. In addition to this, the house power demand is more affected by the weather. Therefore, it is essential to train the load forecast model with both extracted appliances' energy consumption and weather data. Moreover, the potential of deep learning and machine learning-based approaches can be exploited to understand the variability among energy-related variables and reduce the short-term load forecasting error.

#### **4.5. Key contributions**

The variability of house power demand depends on the dynamic behaviors of energy clients. The accurate short forecast at appliances and home levels can help energy consumers for decision making and reducing energy bills.

The accuracy of any forecasting method is highly dependent on the data selection and manipulation. Most of the load forecasting techniques are trained with prior aggregated home data or weather data. Few works of literature have added appliances level data in the

model training but to the best of our knowledge, none has combined the benefits appliances level data with weather information for solving the load forecasting task.

We explored the house and appliances energy consumption to access the dynamic behavior of home residents. We provided individualized energy statistics that illustrate the percentage energy contribution of each important home appliances to the actual home energy consumption.

In terms of data preprocessing, different researches have tried to estimate the appliances signal from the aggregated signal using machine learning or deep learning techniques. However, each literature claims to present the method that is superior to others in terms of disaggregation accuracy. We present and evaluate the ability of the optimization, probabilistic, and deep learning-based network for extracting appliances' features from the aggregated home power demand.

In terms of accuracy of load forecast, as opposed to other techniques that are trained with either prior total power or weather parameters, this thesis combines the benefits of both appliances level data and weather information to forecast the future house energy consumption. Moreover, the thesis evaluates the ability to stack modeling made with long short-term memory (LSTM) and random forest (RF) to reduce the error in the 3 days load forecast.

## CHAPTER 5: EXPERIMENTS USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES FOR LOAD FORECAST AND DISAGGREGATION

This chapter covers the experiments on load forecasting and load disaggregation using machine learning and deep learning techniques. Section 5.1 provides the experiments on load forecasts and performance comparison. Section 5.2 provides the experiment on load disaggregation and performance comparison.

### 5.1 Load forecasting-based techniques

This section covers experiments on forecasting the household power demand. We experimented with random forest-based approaches as well as the support vector machine technique.

Figure 5.1. illustrates the stages that were followed for data preparation, model training, and obtaining the evaluation results. The home energy dataset [8] with 9 features and 3290 observations were cleaned before extracting more important features and splitting into train and testing portions. The values of the features were downscaled between 0 and 1 using Min-max scaler functions available in python. This data normalization was done to make the model function to apply the appropriate weights to all the features. The model training was conducted with data from 1<sup>st</sup> to 19<sup>th</sup> of every month. The hyperparameter tuning based on cross-validation via Grid search was utilized to obtain the optimal parameters within the sliding search space. The testing data contains information from 20<sup>th</sup> to 30<sup>th</sup> of every month were used to evaluate the performance of each load forecasting model.

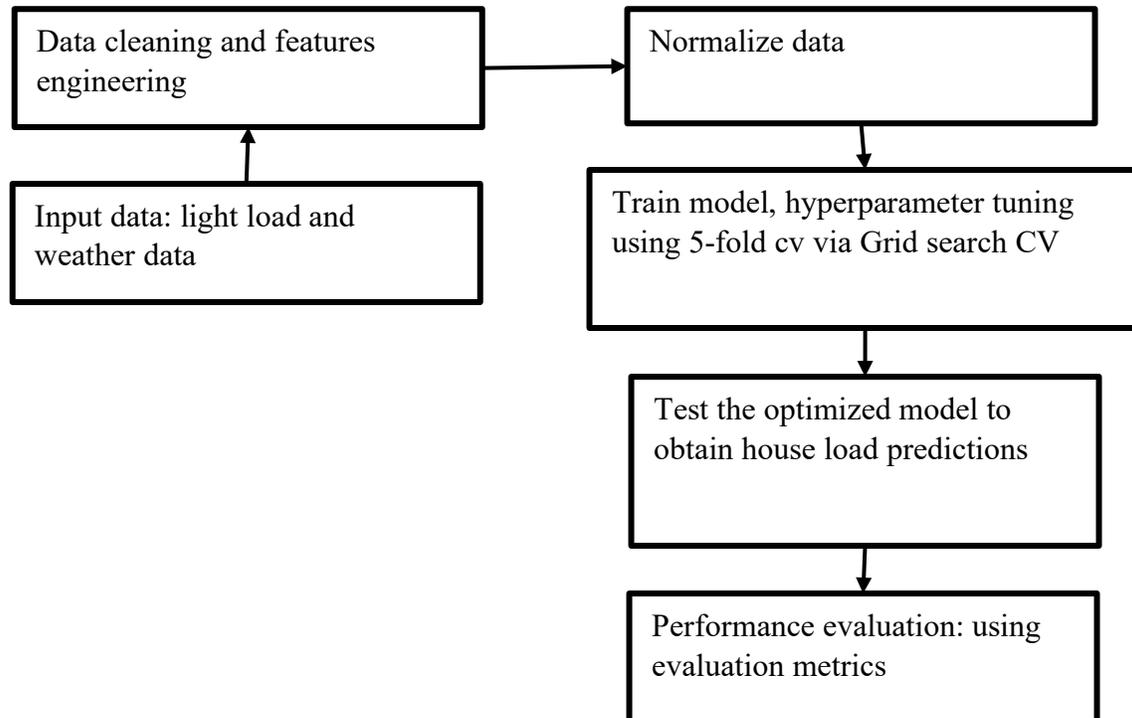


Figure 5.1: Overview of load forecasting procedure

### 5.1.1 Experiment 1: Random Forest (Ensemble Learning), add bagging and boosting (AdaBoost, XGBoost, Gradient Boosting)

The random forest technique uses the decision tree algorithm that constructs the target or predictions based on many observations. In ensemble learning, the generated number of classifiers are summed up to obtain the result. The bagging and boosting are the two random methods used to show the relationship between the previous observations and the results. With the bagging method, the future observation tree is independent of the previous tree whereas, with the boosting method, the final tree is seen as a correction of the previous trees [41,42]. We experimented by tuning the number of trees and depth of trees with the brute force method. The number of trees was in the 1-500 range whereas the depth trees were in the 1-10 range.

The load forecasting experiment with random forest started with a feature engineering based on the python preprocessing OneHotEncoder method that split Date Time into hours of the day.

The model training was conducted with data from 1<sup>st</sup> to 19<sup>th</sup> of every month in the dataset [8]. After gradient boosting model training, the hyperparameter tuning based on cross-validation via Grid search was utilized to obtain the optimal parameter (number of trees=500, the height of trees=1-10) as shown in figure 5.2. The model testing has been done using data from 20<sup>th</sup> to 27<sup>th</sup> of every month. The implementation of this experiment of the random forest for forecasting the house energy consumption was adapted from [44,45].

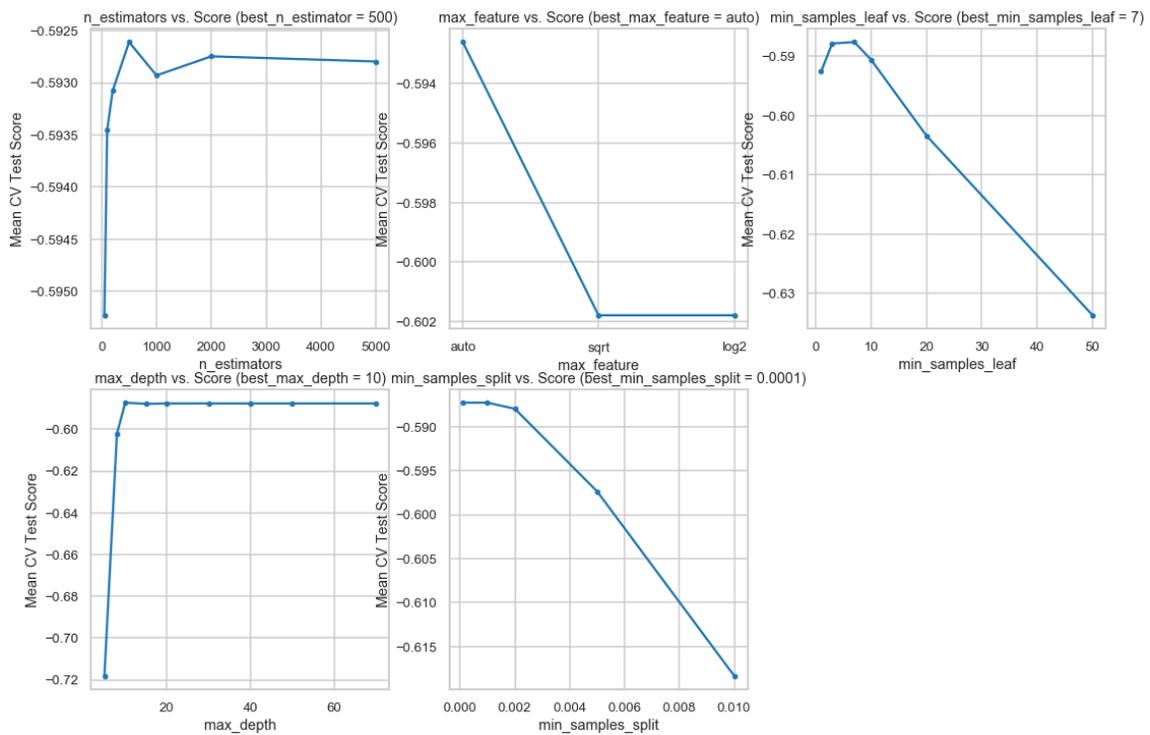


Figure 5.2:Hyperparameter tuning for random forest with gradient boosting method

The accuracy metrics used in the load experiment are defined as follows [3,34]. The mean absolute error (MAE) is defined as a metric that helps to evaluate the convergence between the actual house power demand and the predicted house power demand. The root mean square error (RMSE) allows the term by term comparison to evaluate the variability of predicted energy consumption.

$$MAE = \frac{\sum_{t=1}^T |\hat{y}^{(i)}(t) - y^{(i)}(t)|}{T} \quad (5.1)$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^T (\hat{y}^{(i)}(t) - y^{(i)}(t))^2}{T}} \quad (5.2)$$

Where  $\hat{y}^{(i)}(t)$  and  $y^{(i)}(t)$  denote the actual and estimated power of an appliance  $i$  at an instant time  $t$ , and  $T$  is the forecasting period.

As shown in figure 5.3, we can see that the household prediction shows low fluctuation for 3 days ahead and is in line with the actual electricity consumption in the household for a time between March 17 and March 19, 2016. The wide mismatch between the random forest prediction and the actual power demand is observed on March 19 and March 20, 2016, and this is originated from the electrical fault in appliances power. Interestingly, the predicted and actual power were in good agreement after March 20, and March 21, 2016.

The contribution of each individual feature to the model performance is explained by the sequence of features important in the GradientBoostingRegressor as shown in figure 5.4. The inside temperature and humidity inside the house play a medium role in influencing the model performance. In terms, a time of weather data, the outside pressure, humidity, and temperature play a great part compared to the inside ones. In terms of energy use, evening hours (between 18 and 20 h) are more significant than morning hours, and this in

line with the fact that many peoples are back from work and more appliances are being used including cooking.

Table 5.1 and figure 5.5 illustrate the results obtained with ensemble learning with simple random forest, gradient boosting, and Adaboost for the same dataset. In terms of forecasting error, the random forest with gradient boosting has resulted in low forecasting error (RMSE= 0.123) compared to 0.125 of simple random forest regressor and 0.14 of Adaboost regressor. The comparative low forecasting error for a model with a gradient boost could be originated from its ability to selected optimal parameters and construct future predictions from the previous observation.

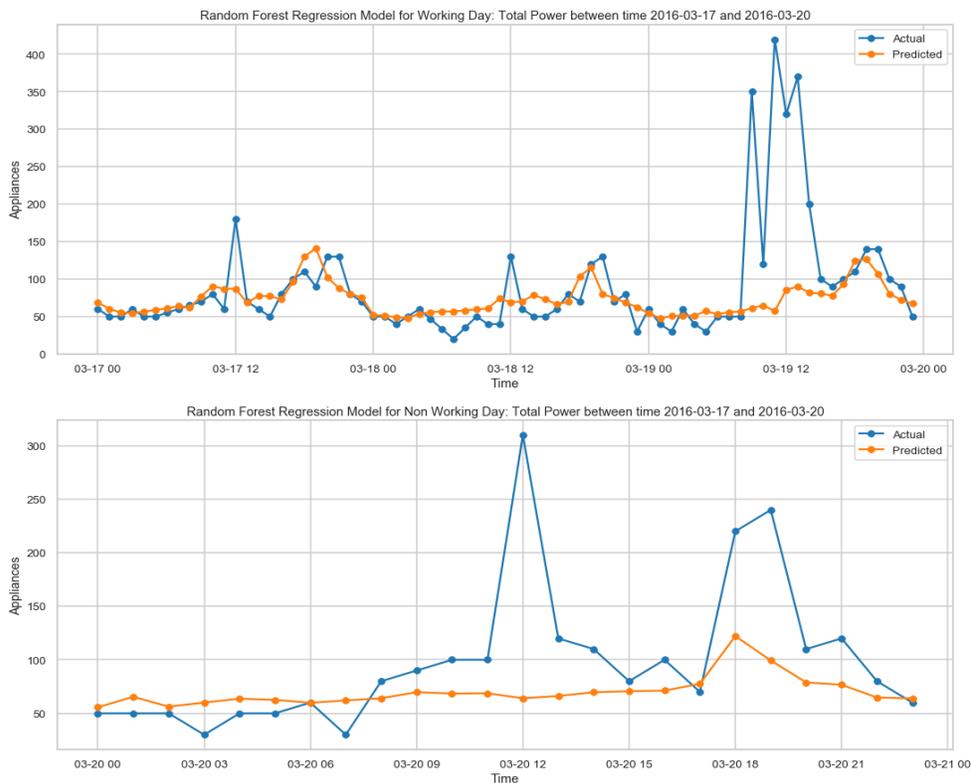


Figure 5.3: Random forest predicted vs actual power demand between Mach 17 and March 21,2016.

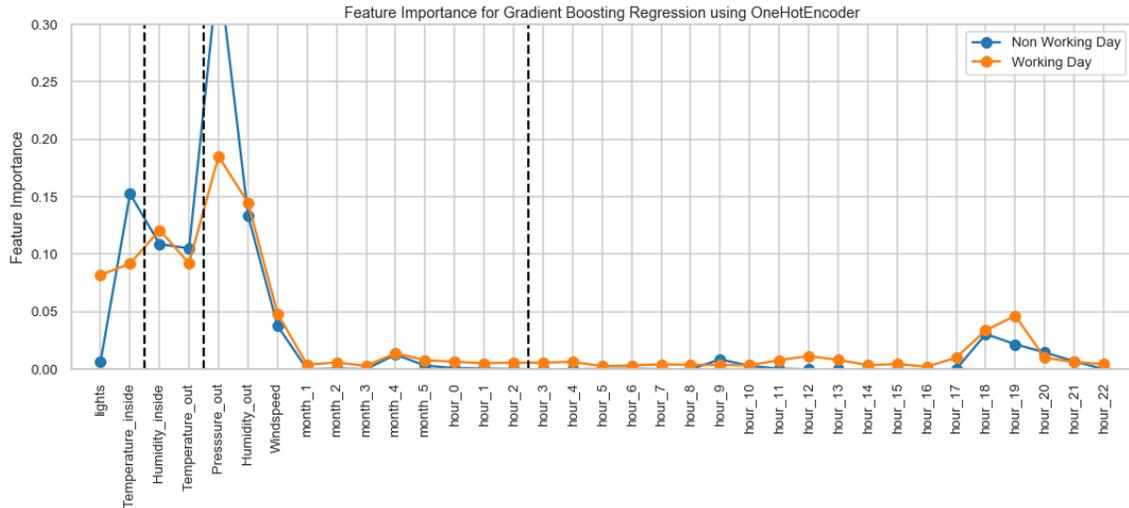


Figure 5.4: Features importance in forecasting performance of GradientBoostingRegressor.

Table 5.1: RF model for load forecast.

RF Approach	MAE	RMSE
<b>RandomForestRegressor</b>	0.095	0.125
<b>GradientBoostingRegressor</b>	0.067	0.123
<b>AdaboostRegressor</b>	0.13	0.14

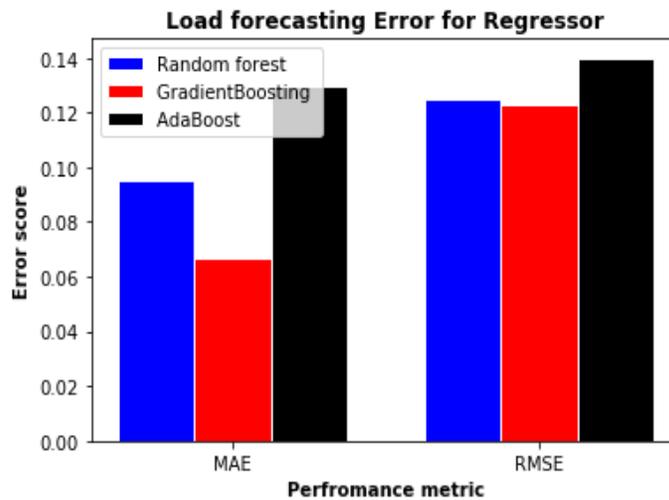


Figure 5.5: RF model for load forecast

### 5.1.2 Experiment 2: Support Vector Machine (SVM) for load forecasting

The support vector regression is an extension of the support vector machine used for designing the time series model. It uses kernel parameters to obtain the input parameters that optimize the model performance [26,36]. We used the radial basic function (RBF) kernel to turn the gamma and cost C parameters.

We used the python processing module to call and train the dataset with the optimized parameters by tuning the kernel parameters: The kernel parameters were tuned as follows. The sigma parameters were in range 0.1 and 5 whereas the cost parameters were in the range of 1 and 100. The optimized parameters for SVR were (kernel='rbf', C=100, gamma=0.1). The implementation of this experiment of the support vector regression for forecasting the house energy consumption was adapted from [43,44].

As shown in figure 5.6 we can see that the 6 days ahead of predicted house power demand shows low fluctuation and is in line with the actual house power demand for a time between May 01 and May 06, 2016. The predicted and actual house power demand was in good agreement for almost 5 days before May 06, 2016, but the large mismatch happened on May 06, 2016, and. As shown in table 5.2 and figure 5.7 the SVR performed well with 92.99% of forecasting accuracy and low forecasting error (RMSE=0.070).

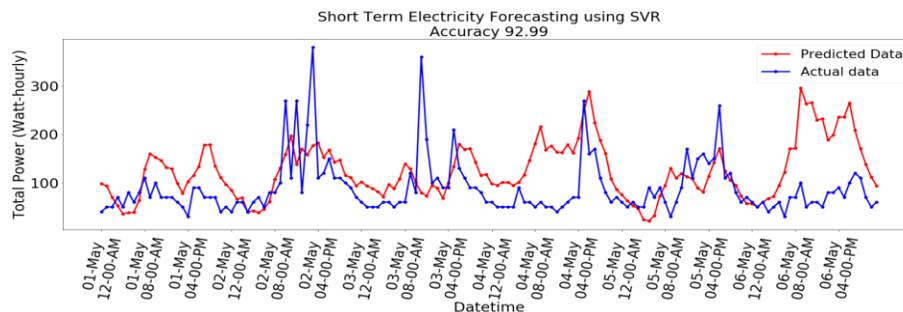


Figure 5.6: SVR predicted vs actual power demand between May 01 and May 06,2016.

Table 5.2: SVM model performance for load forecast.

SVM Approach	MAE	RMSE
<b>SVR</b>	0.046	0.070

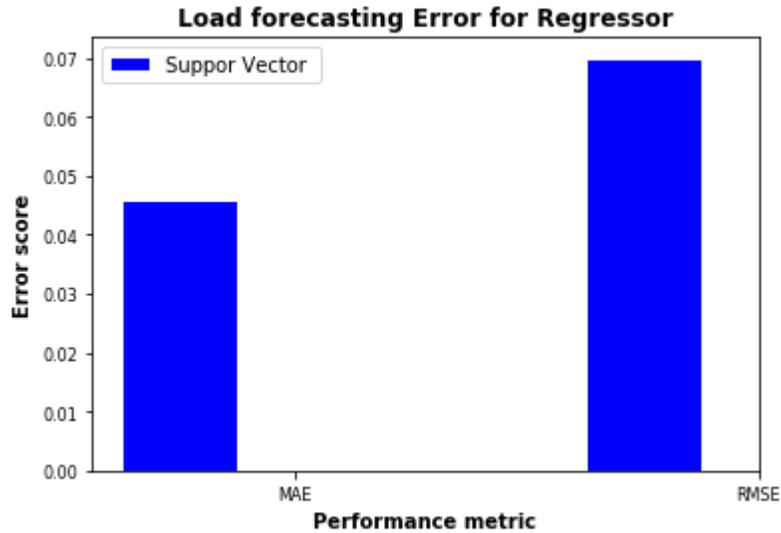


Figure 5.7: Forecasting accuracy of SVR

### 5.1.3 Performance comparison of load forecasting approaches

The performance comparison among load forecasting methods is shown in Table 5.3 and figure 5.6. It is clearly shown that the SVR outperforms other 3 load forecasting methods. In terms of root mean square error (RMSE), the SVR achieved 0.070 compared to 0.123 of random forest with gradient boosting and 0.14 of random forest with AdaBoost.

Table 5.3: Performance comparison for regression-based load forecasting approaches

Load forecasting Approach	MAE	RMSE
<b>RandomForestRegressor</b>	0.095	0.125
<b>GradientBoostingRegressor</b>	0.067	0.123
<b>AdaboostRegressor</b>	0.13	0.14
<b>SVR</b>	0.046	0.070

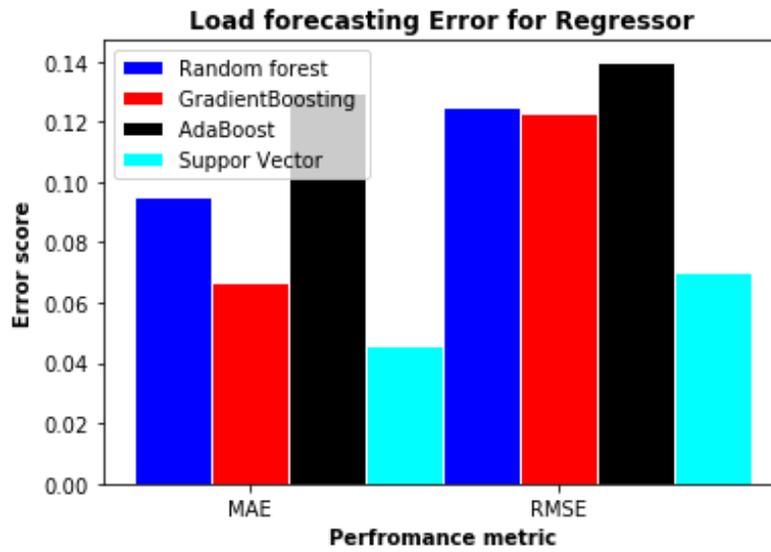


Figure 5.8: Performance comparison for regression-based load forecasting approaches

## 5.2 Load disaggregation

The load disaggregation experiments cover the supervised and unsupervised techniques that separate the aggregated house power demand with the individual appliances' power demand. In this section, we applied combinatorial optimization (CO), probabilistic approach, and four deep learning-based approaches. The models' architecture and a comparison of disaggregation results have been discussed.

### 5.2.1 Experiment 1: Combinatorial Optimization (CO) for Load Disaggregation

The information, architecture, and training process of combinatorial optimization are presented in this section.

#### A) CO model architecture

In practical load disaggregation, the combinatorial optimization (CO) works on supervised learning which requires the signature libraries of appliances to train the model. The CO technique is designed and trained to minimize the error (i.e. noise term) between the

aggregated house power demand and the sum of all considered individual appliance energy consumption. To solve this minimization problem, it is required to design the CO model that limits the contributions of other no essential appliances or appliances with the same power consumption in the house.

The objective function is defined as the minimization of the difference between the aggregate and sum of appliances power demand and is formulated as follows [6,40].

$$\min \left| \hat{y}(t) - \sum_{i=1}^N y^{(i)}(t) \right| \quad (5.3)$$

Where  $\hat{y}(t)$  is the aggregated household power demand,  $y^{(i)}(t)$  is the power demand of appliance  $i$ , and  $N$  is the number of appliances that are present in the house.

The CO model along with the Factorial Hidden Markov model (FHMM) are the two benchmark load disaggregation algorithms that are used to evaluate the disaggregation performance of other NILM approaches.

## **B) CO model training on Reference Energy Disaggregation Dataset (REED)**

The training process of the CO approach for this experiment is described as follows: The REED dataset was first converted into the hierarchical data format (HDF5) to allow data analysis using a non-intrusive load monitoring toolkit (NILMTK) [14]. The NILMTK toolkit has been written using python and contains features that enable dataset conversion, along with different functions for data manipulation, data extraction, and presenting energy proportion and statistics. The CO model was trained using appliance level data and the trained model was used to disaggregate the aggregated house power demand to obtain the state and energy consumption of each of the 5 important appliances present in the first

building of REDD dataset. Finally, different metrics were used to evaluate the disaggregation performance of combinatorial optimization. The implementation of this experiment of the combinatorial optimization for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

### C) Model Performance Evaluation metrics

We used several metrics to evaluate the performance of load disaggregation-based techniques. The metrics were used to evaluate how the model can classify and detect the appliances event from aggregated house power demand.

Referring to the ON/OFF type of the appliances, we can define the metrics as follows:

*TP*(True positive): Shows the ON state appliance that is detected to be in ON state by the model

*FP*(False positive): Shows the OFF-state appliance that is detected to be in ON state by the model.

*TN*(True negative): Shows the OFF-state appliance that is detected to be in OFF state by the model.

*FN*: (False negative): Shows the ON-state appliance that is detected to be in OFF state by the model.

*P*: Sum of positives in the actual energy consumption of an appliance

*N*: Sum of negative in the actual energy consumption of an appliance

$y^{(i)}(t)$  = the actual energy consumption of an appliance  $i$  at the instant time  $t$

$\hat{y}^{(i)}(t)$  = the estimated energy consumption of an appliance  $i$  at the instant time  $t$

Taking example of lighting load, the following metrics can be defined [3,4].

Recall: Indicates proportion of the energy that is was correctly detected by the model.it

$$Recall = \frac{TP}{TP + TN} \quad (5. 4)$$

Precision: Indicates proportion of the total energy that is assigned and consumed by an appliance.

$$Precision = \frac{TP}{TP + FP} \quad (5. 5)$$

*F1 score* combines the precision and recall.

$$F1 \text{ score} = 2 \times \frac{Precision * Recall}{Precision + Recall} \quad (5. 6)$$

Accuracy indicate the generalization score:

$$Accuracy = \frac{TP + TN}{P + N} \quad (5. 7)$$

*MAE*: indicates the relative error in the energy attributed to every appliance at a time  $t$ .

$$MAE = \frac{\sum_{t=1}^T |\hat{y}^{(i)}(t) - y^{(i)}(t)|}{T} \quad (5. 8)$$

$$\text{Relative error in total energy} = \frac{|\hat{E} - E|}{\max(\hat{E}, E)} \quad (5. 9)$$

The disaggregated power signals for 5 important appliances present in house1 of the REDD dataset are shown in figure 5.9. The dishwasher and electric oven consume more power for a given period, but they are found to be in OFF state for most of the time. As shown in

Table 5.4 and figure 5.10, the CO approach performed well for the fridge and sockets but fails for other appliances.

As shown in figure 5.9, the CO approach was able to estimate the electricity consumption of 5 important appliances based on the aggregated home power demand. On April 18, 2011, a dishwasher and electric oven consumed more electricity compared to microwaves and sockets, but they were being used for a very limited time during the day. In terms of load disaggregation performance, the results in figure 5.10 and table 5.7 illustrate that the CO approach performed well for the fridge and electric oven but showed high disaggregation error on microwave and washer-dryer. The lowest root mean square error (RMSE) was found to be 168.3 for microwave whereas the highest (RMSE=312) error is obtained for microwave.

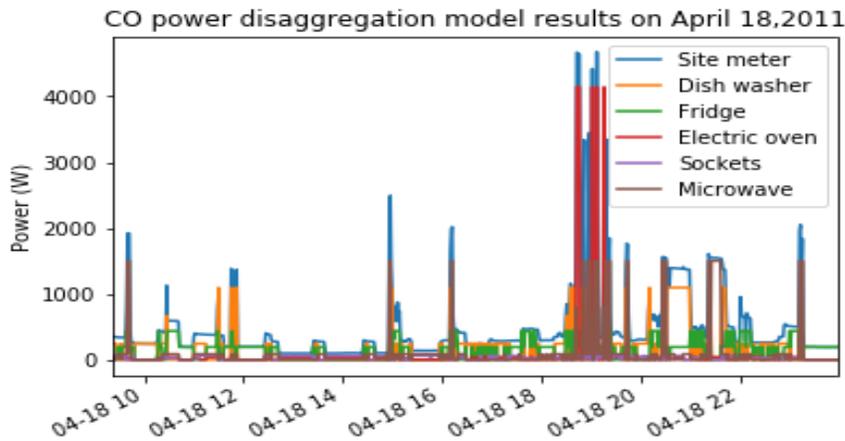


Figure 5.9: Appliance power disaggregated from site meter with CO approach

Table 5.4: CO energy disaggregation performance for home 1 seen during training

Combinatorial Optimization performance on house 1		
	F1 score	RMSE
Fridge	0.42	168.3
Microwave	0.048	312
Washer dryer	0.20	280
Electric Oven	0.022	239

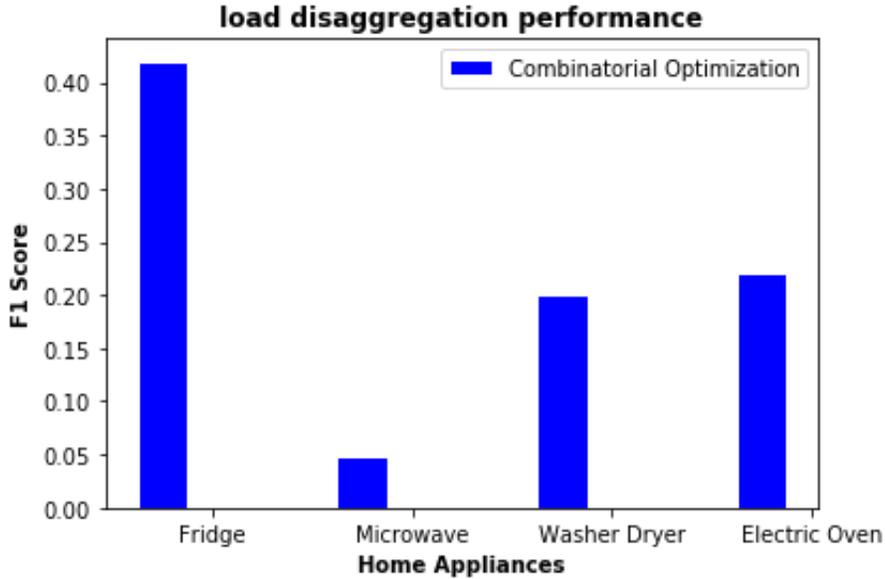


Figure 5.10: F1 score obtained with CO for home 1 seen during training

### 5.2.2 Experiment 2: Factorial Hidden Markov Model (FHMM) for Load Disaggregation

The factorial hidden Markov (FHMM) model is a kind probabilistic technique that is constructed based on the Markov process to model the time series data. The FHMM encodes prior information into a single multinomial variable (i.e. hidden state) with  $K$  discrete state values. During the modeling of the FHMM, prior appliances consumptions are modeled by specifying the relation between the observations and a sequence of hidden states and Markov transition linking the hidden states. In practical load disaggregation, the hidden Markov model links the observations and hidden states so that at any instant  $t$  of a period  $T$  the model can break the aggregated house power demand  $\hat{y}(t)$  into different appliances  $y^{(i)}(t)$  [6,45].

We used the NILMTK [14] toolkit functionalities to visualize the appliances states and data splitting before call the FHHM model and train with the appliances level data. The

only considered the 5 appliances with a significant impact on house power demand and the training duration cover the appliances' signal of 10 days of REDD dataset. The implementation of this experiment of the FHMM for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

The disaggregated power consumption of 5 important appliances and the aggregated home power demand (site meter) that are present in house 1 in the REDD dataset are shown in figure 5.11. The dishwasher and electric oven consumed more electricity in a limited short period, but they were found to be in OFF state for most of the time on April 18, 2018.

In terms of load disaggregation performance, the results in figure 5.12 and table 5.5 illustrate that the FHMM approach performed well for the fridge and electric oven but showed high disaggregation error on microwave and washer dryer. By comparison, the lowest root mean square error (RMSE) was 138 for microwave whereas the highest (RMSE=301.7) error is obtained for microwave.

The disaggregation performances of two load disaggregation benchmark approaches (CO and FHMM) were compared in figure 5.8(b) and it is clear the combinatorial optimization performance worse in disaggregating all 5 important appliances.

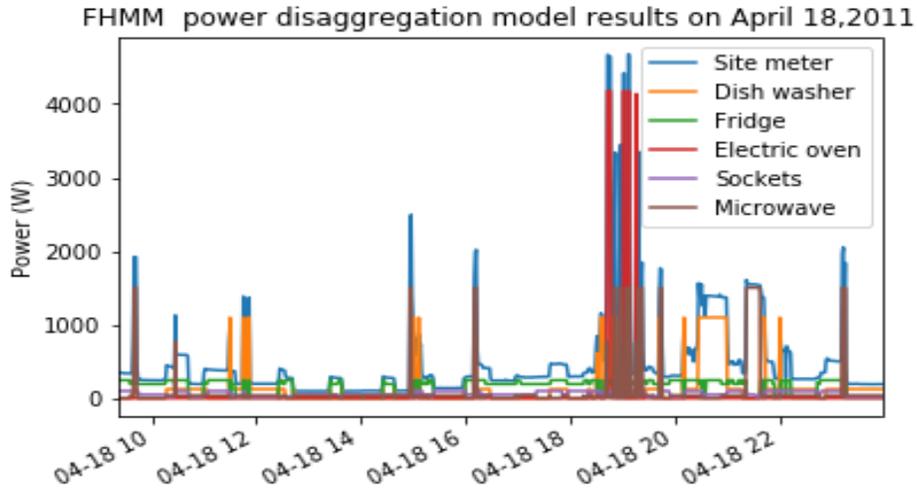


Figure 5.11: Appliance power disaggregated from site meter with FHMM approach

Table 5.5: FHMM energy disaggregation performance for home 1 seen during training

Factorial Hidden Markov Model (FHMM) performance on house 1		
	F1 score	RMSE
Fridge	0.52	138
Microwave	0.02	301.7
Washer dryer	0.24	260.2
Electric Oven	0.57	240

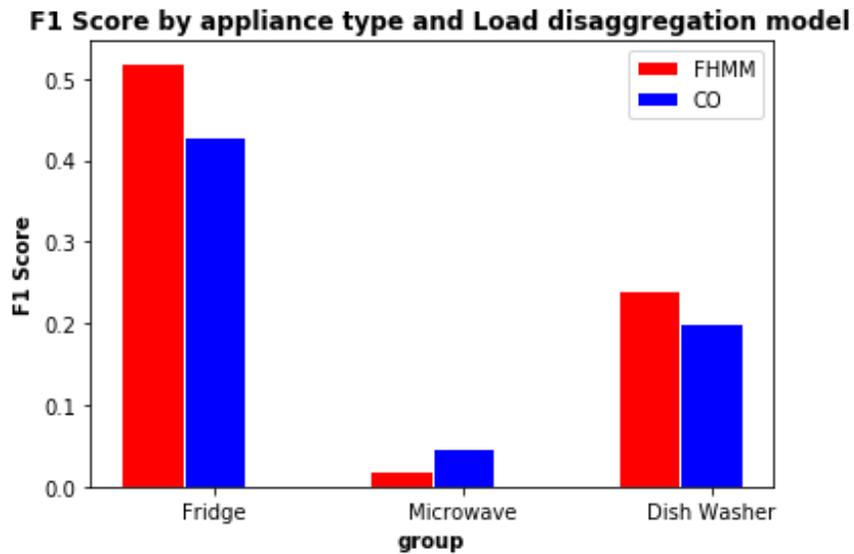


Figure 5.12: F1 Score performance comparison between CO and FHMM approach

### 5.2.3 Experiment 3: Denoising Autoencoder (DAE) Neural Network for Load Disaggregation

#### A) DAE model Architecture and training on Reference Energy Disaggregation Dataset (REED)

The denoised autoencoder (DAE) is a kind of autoencoder network in which the number of inputs is greater than the hidden units. The DAE is aimed to learn from the noisy inputs and achieve the non-noisy output target.

During the DAE training, the corrupted signal from aggregated house power demand is used to train the network before it is fed with the target appliance level data in the testing stage. The aggregated house power demand is found to be corrupted because it contains signals of other non-essential appliances that are not under consideration during the model training [3,4].

The noise signal  $x'$  is mapped with an autoencoder to obtain the following hidden representation [4].

$$h(x') = f(W_1'x' + b_1') \quad (5.9)$$

Then, the clean signal  $y$  is reconstructed from that hidden representation  $h(x')$  as follows.

$$y = f(W_2'h(x') + b_2') \quad (5.10)$$

Where  $W_1'$ ,  $W_2'$  are the weight of the noisy and clean signal, whereas  $b_1'$  and  $b_2'$  are the bias vector of the noisy and clean signal.

The DAE network was designed to contain five layers stacked together [4]. The first layer of this network was 1 dimensionally convolutional with 8 filters of size 4, followed by the 2 layers that are fully connected with 128 units each, one fully connected layer with 120

units and last with convolution layer with 9 filters of size 4. The reLu activation was used to transform the output obtained from each layer as the input to the next layer.

The number of layers and neurons were selected using the manual search that selects the hyperparameters from the search space, used them to train the model and calculate the disaggregation error. The hyperparameter search is repeated many times until the search space is completed, and we selected the parameter that helps the model to low disaggregation error. The search space number of layers was from 1 to 10 whereas the number of neurons for each layer was selected from 1 to 220 range. Adam optimizer was utilized to optimize the DAE network and a learning rate of 0.001 was chosen. The model was trained in a batch size of 16 and with 25 epochs. The implementation of this experiment of the DAE network for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

The estimated power demand of some important appliances and the ground truth related to those appliances that are present in house 1 is shown in figure 5.13. The variation trend of the estimated power of the fridge and microwave is in line with the variation of the actual power (ground truth) on April 30, 2011.

With regards to the disaggregation error, results in figure 5.14 and table 5.6 indicate that the DAE approach achieved good disaggregation results for the fridge, light, and microwave whereas low performance is obtained for electric oven. The highest F1 score was 0.94 for light whereas the lowest score (0.32) is obtained for electric oven.

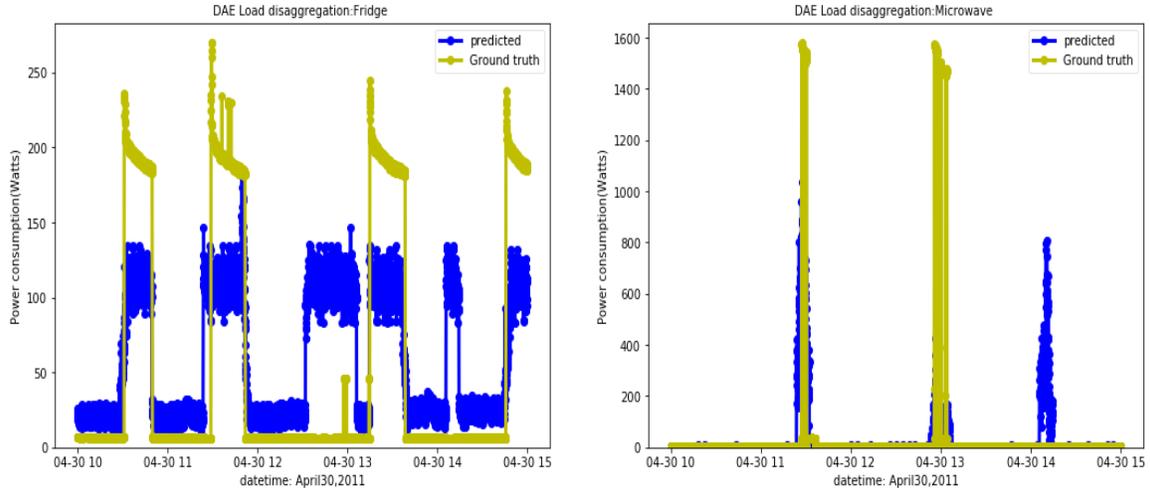


Figure 5.13: DAE estimated vs ground truth power demand of fridge and microwave in house1

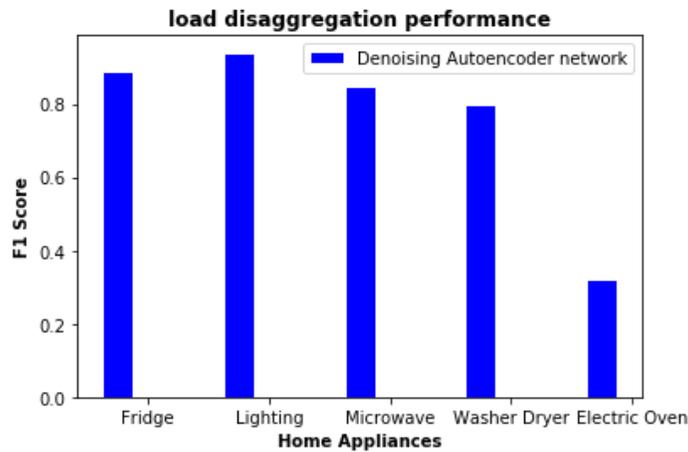


Figure 5.14: DAE disaggregation F1 score for different appliances

Table 5.6: DAE Energy disaggregation performance for home seen during training

Denoising Autoencoder (DAE) neural network						
	Recall	Precision	F1 score	Accuracy	Relative Error(watt)	MAE(Watts)
Fridge	0.99	0.80	0.89	0.80	0.68	41
Light	0.99	0.84	0.94	0.89	0.73	38
Microwave	0.99	0.74	0.85	0.74	0.87	33
Washer dryer	1	0.67	0.80	0.67	0.38	28
Electric Oven	0.00019	0.75	0.32	0.00023	0.27	11

#### **5.2.4 Experiment 4: Recurrent Neural Network based Long Short-Term Memory (LSTM) for Load Disaggregation**

The recurrent neural network is a type of network that can memorize the previous information and use that information to learn the current data. It is found to be suitable for solving the problem that involves the sequence of data such as the time series home energy consumption. In the recurrent network, the decision at the time  $t$  is based on the decisions made from the previous time step  $t - 1$ . Although the recurrent network performed well with dealing with sequence data, it faces challenges with training the long length data as gradients tend to vanish or become large if the network must go in multiple layer networks. The long short-term memory (LSTM) is a kind of recurrent network which can keep its memory state during the sequence cycle. The internal state of LSTM communicates with the inputs and previous output to determine what inputs to maintain, update, and delete. It is made with the input gate that maintains its inner state; the forget gate which helps in filtering the previous inputs; the output gate for deciding the data to feed in the LSTM internal state[47,48].

The LSTM architecture is described as follows [4]:4 layers were stacked on the first layer during LSTM training. The first layer of this network was Convolutional with 8 filters of size 4, followed by the 2 layers that are bidirectional LSTM with 128 units each; the third layer was the fully connected with 258 neurons followed with the fully connected layer with linear activation. The hyperbolic tangent activation was used to transform the output obtained from each layer to the next layer. To avoid the model overfitting, the 10% dropout was added for the input layer. The mean absolute error was chosen for as the loss function whereas the Adam optimizer with learning rate 0.001 was utilized. The implementation of

this experiment of the LSTM network for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

The estimated power demand of two (microwave and fridge) important appliances and the ground truth related to those appliances that are present in house 1 is shown in figure 5.15. The variation trend of the estimated power of the fridge and microwave follows the variation of the actual power (ground truth) on April 30, 2011.

The disaggregation performance of LSTM is shown in figure 5.16 and table 5.7. The results indicate that the LSTM approach achieved good disaggregation results for light, microwave, and fridge whereas low performance is obtained for electric oven and washer dryer. This could be explained by the fact that the LSTM network can easily disaggregate appliances with simple signature features compared to that of complex or multi-state signature features. In terms of the F1 score, the LSTM succeeded with the disaggregating signal feature of lighting with 0.89 and 0.64 for the fridge but failed to disaggregated washer-dryer with 0.058. Obtaining good disaggregation results for both ON /OFF state types such as light and multi-state types of an appliance such as fridge can be seen as a good achievement for recurrent based networks.

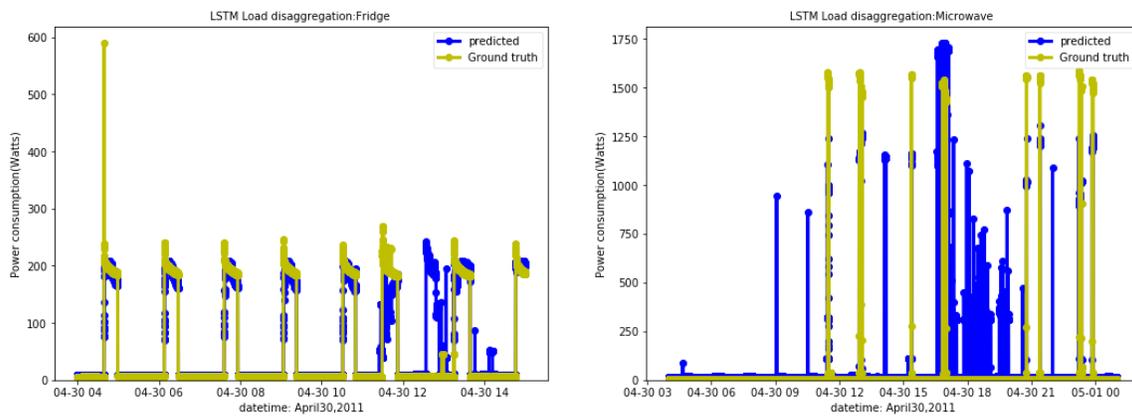


Figure 5.15: LSTM estimated vs ground truth power demand of fridge and microwave in house1

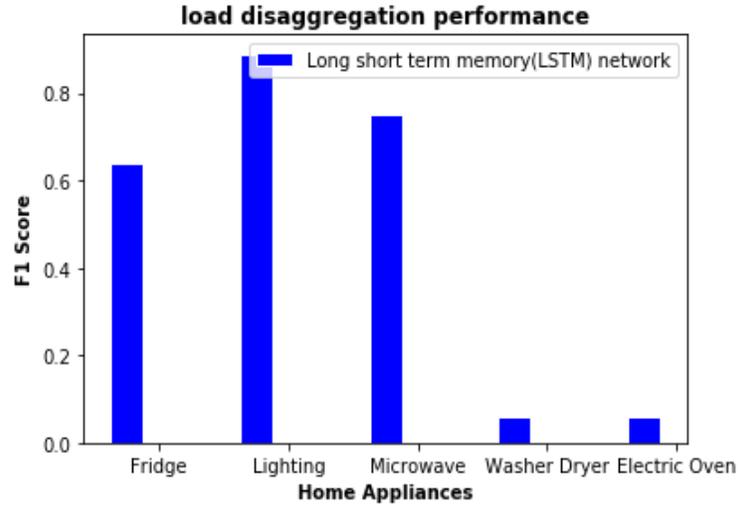


Figure 5.16: LSTM disaggregation F1 score for different appliance

Table 5.7: LSTM energy disaggregation performance analysis for home 1 seen during training.

Long short-term memory (LSTM)						
	Recall	Precision	F1 score	Accuracy	Relative Error	MAE
Fridge	0.48	0.88	0.64	0.53	0.71	43
Light	0.61	0.92	0.89	0.81	0.75	39
Microwave	0.77	0.74	0.75	0.63	0.87	32
Washer dryer	1	0.92	0.058	0.34	0.63	39
Electric Oven	0.03	0.75	0.059	0.34	0.50	16.88

### 5.2.5 Experiment 5: Gated Recurrent Unit (GRU) Neural Network for Load Disaggregation

The gated recurrent unit (GRU) is a kind of recurrent network that shares similarity (i.e. gate) with the LSTM network but differ to LSTM on the fact that it exposes its entire memory to the inputs data. The GRU contains the update that determines the memory space to be exposed to the input and the reset gate that allows new inputs to be fed in the previous memory [49].

The GRU network was designed as follows [4]. Five layers were stacked on the first layer during GRU training. The first layer of this network was Convolutional with 8 filters of size 4, which was followed by two bidirectional GRU layers (one with 64 neurons and other with 128 neurons). The third layer was the fully connected layer with 64. the reLU activation was used to transform the output obtained from each layer as the input to the next layer. The final layer was Fully connected. The linear activation has been selected. The mean absolute error was chosen for as the loss function whereas the Adam optimizer with learning rate 0.001 was utilized.

The model was trained with few appliances level data to disaggregate the aggregated house power demand and extract state and energy signal of each of the 5 important appliances in the house. The implementation of this experiment of the GRU network for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

The forecasted and actual power demand of the fridge located in house 1 is shown in figure 5.17. The variation trend of forecasted power of the fridge agrees with the actual power (ground truth) on some date time of April 30, 2011.

The performance of GRU architecture is shown in figure 5.18 and table 5.8 As has been seen with other recurrent based networks, the GRU architecture achieved good disaggregation results for light but it showed high disaggregation error for a microwave. The highest F1 score was 0.90 for the washer-dryer whereas the lowest score (0.52) is obtained for the fridge. As opposed to the LSTM architecture that succeeded with disaggregation appliances with simple signature features such as lighting and microwave, the GRU network succeeded in disaggregating the appliances with a complex signature

such as the washer dryer. This could be explained to the ability of the GRU network to expose its entire memory to the input data.

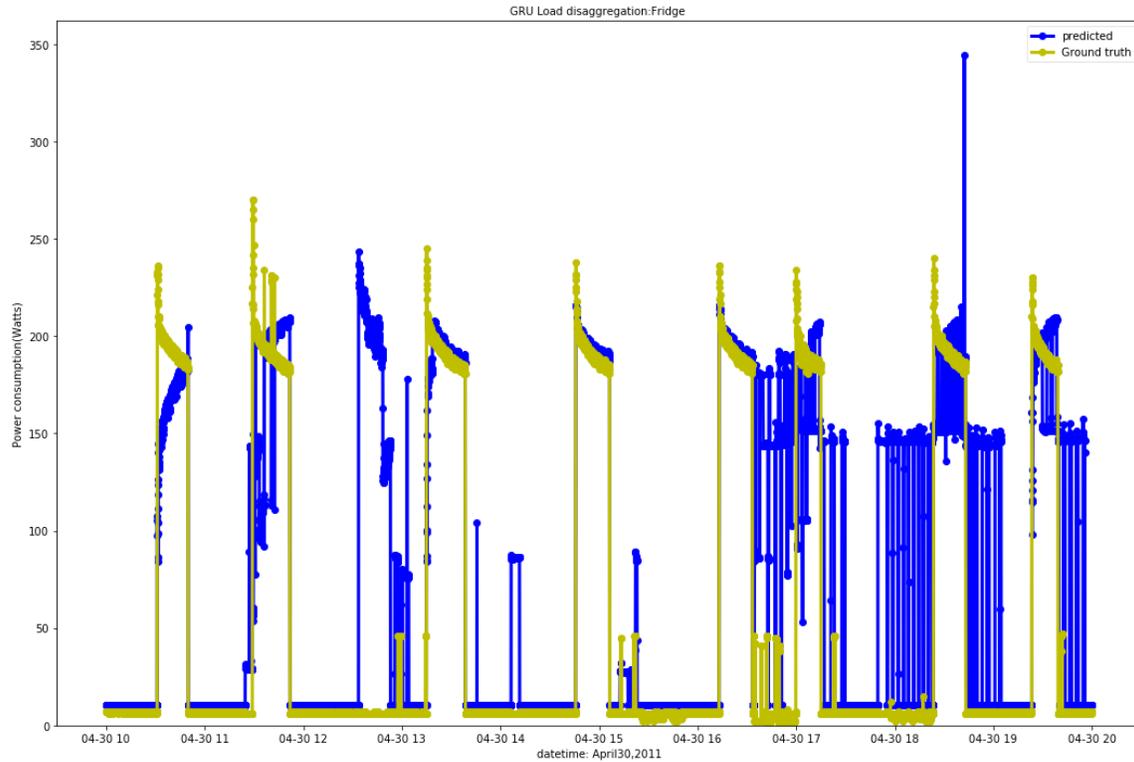


Figure 5.17: GRU estimated vs ground truth power demand of fridge and microwave in house1

Table 5.8: GRU Energy disaggregation performance for home 1 seen during training.

Gated Recurrent Unit neural network performance on House 1						
	Recall	Precision	F1 score	Accuracy	Relative Error	MAE
Fridge	0.35	0.95	0.52	0.46	0.72	43
Light	0.61	0.94	0.74	0.62	0.61	27
Microwave	0.037	0.81	0.071	0.27	0.84	25
Washer dryer	1	0.67	0.80	0.67	0.67	46
Electric Oven	1	0.67	0.80	0.67	0.73	28

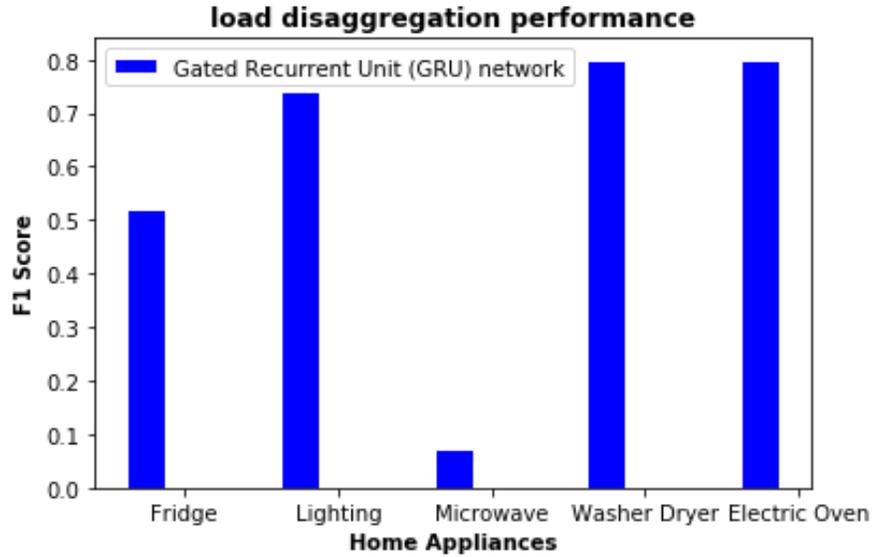


Figure 5.18: GRU disaggregation F1 score for different appliance

### 5.2.6 Experiment 6: Convolution based Seq 2-point Neural Network for Load Disaggregation

The architecture and training process of sequence to point network are presented in this section.

The sequence to point network differs from other neural networks with its way of representing the midpoint of an appliance's window instead of considering the entire input window. For example, if the main window is represented by the input  $Y_{\tau:t+w-1}$ , the sequence to point network will be trained with the midpoint element to represent the targeted appliance with the output at  $\tau = t + w/2$ . In the training process, the state of the midpoint element for the target appliance is related to the main information (i.e. aggregated house power demand). The loss function that is required to train the seq2 point neural network is represented by [10].

$$L_{P=} \sum_{i=1}^{t-w+1} \log P(x_{\tau} | Y_{t:t+w-1}, \theta) \quad (5.11)$$

Note that both at the start and end of the sequence to point training, the factor  $w/2$  is equated to zero.

The model was designed to have 7 layers stacked together: The first 5 layers were convolution type with neurons 10,30,40,50,50, and with size between 1 and 6. The reLU activation was used to transform the output obtained from each unit layer as the input to the next layer. The next two layers were fully connected with 50 and 1024 neurons for each layer.

In this experiment, the model is trained with the aggregated house and the appliance level data to predict the midpoint values which correspond to the target appliance window. The implementation of this experiment of the sequence to point network for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14].

The forecasted and actual power demand of the fridge and microwave located in house 1 is shown in figure 5.19. The variation trend of forecasted power of the fridge is in line with the actual power (ground truth) on some date time of April 30, 2011. The wide changes in projected power demand are observed for a microwave.

The performance of sequence to point architecture is shown in table 5.9. A sequence to point-based network achieved good disaggregation results for the fridge and electric oven, but it showed a high disaggregation error for a microwave and washer dryer. The highest

F1 score was 0.80 for the electric oven whereas the lowest score (0.067) is obtained for the washer dryer.

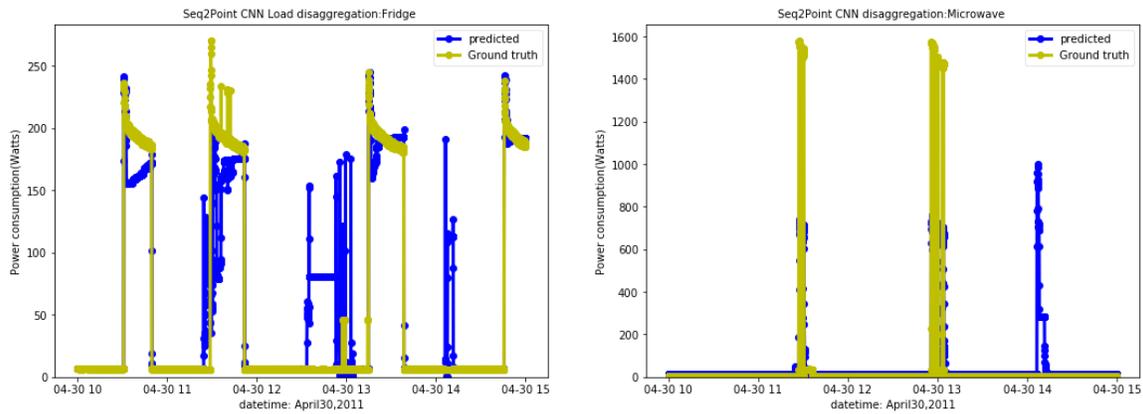


Figure 5.19: CNN Estimated vs ground truth power demand of fridge and microwave in house1

Table 5.9: Conv sequence to point Energy disaggregation performance analysis for home seen during training.

Conv sequence to point neural						
Appliance type	Recall	Precision	F1 score	Accuracy	Relative Error(watts)	MAE (watts)
Fridge	0.34	0.97	0.50	0.46	0.72	42
Microwave	0.99	0.80	0.85	0.74	0.84	28
Washer dryer	0.03	0.90	0.067	0.34	0.38	22
Electric Oven	1	0.67	0.80	0.67	0.71	28

### 5.2.7 Performance comparison of load disaggregation-based approaches

Performance comparison of 6 load disaggregation-based approaches is shown in table 5.10 and figure 5.20. The benchmarks algorithms (CO and FHMM) succeeded with a finite state appliance (e.g. fridge) but failed to disaggregate continuously varying or multistate appliances such as the washer dryer. In addition to this, the performance is affected by the training data (number of appliances). For example, by reducing the number of the appliance

from 20 to 5, the CO disaggregation score for the fridge is increased from 53 to 69%. Comparatively, the denoising autoencoder network out-performs the benchmark NILM algorithms (CO and FHMM) and achieved an F1 score of 89% on the fridge and 80% on washer dryer. The sequence to point architecture was able to accurately disaggregate the microwave (F1 score=0.85) and electric oven (F1 score=0.80) signals.

Table 5.10: F1 score comparison of load disaggregation-based approach tested on home1 seen during training.

F1 score of loads disaggregation-based approaches: number of appliances in the model n=20							F1 score: n=5
	CO	FHMM	DAE	LSTM	GRU	Seq to point	
Fridge	0.53	0.74	0.89	0.46	0.52	0.51	0.69
Washer dryer	0.10	0.24	0.80	0.058	0.80	0.067	
Microwave	0.20	0.02	0.85	0.75	0.071	0.85	0.35
Electric oven	0.13	0.57	0.32	0.059	0.80	0.80	

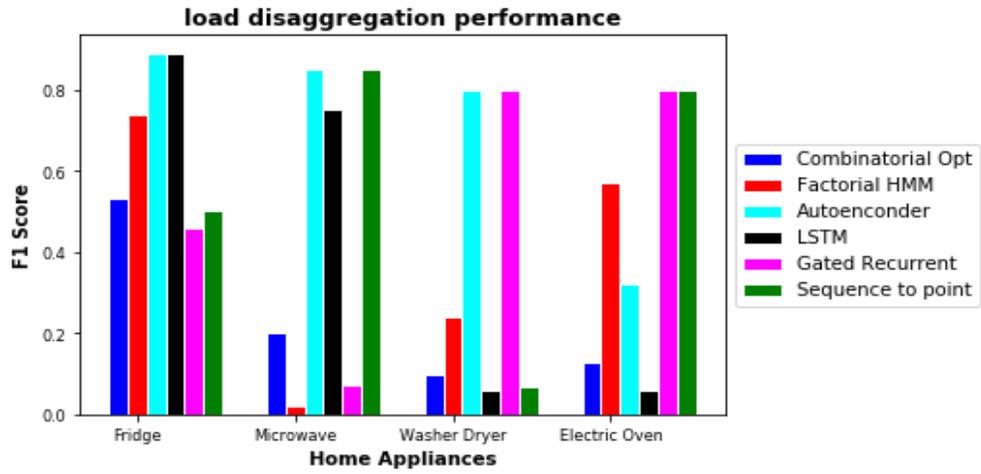


Figure 5.20: F1 score comparison of load disaggregation-based approach tested on home1 whose data seen during training.

## CHAPTER 6: PROPOSED SOLUTION

This chapter discusses the development of the proposed load forecast based on load disaggregation and ensemble learning. The appliances' signal patterns are obtained with load disaggregation and these represent the dynamic behavior of home residents. As shown in figure 6.1, we used the combinatorial optimization (CO) to disaggregate signal features of 7 individual appliances from the aggregated energy consumption in the house. The CO based on pattern similarities was designed to identify repetitions and patterns with similar shapes in house and appliances level data to achieve unification and reduce the complexity of disaggregation tasks.

Since the house's energy consumption is affected by both the appliances energy and the meteorological weather, the proposed forecast is designed to learn from estimated energy consumption of 7 appliances and weather information to predict the future house energy consumption. The load forecast architecture made with an ensemble of long short-term memory (LSTM) and random forest regression (RF) was utilized to forecast the house energy consumption for 3 days ahead. The three months appliance level and weather parameters were used to train the LSTM-RF ensemble whereas the one-month data has been used for model testing.

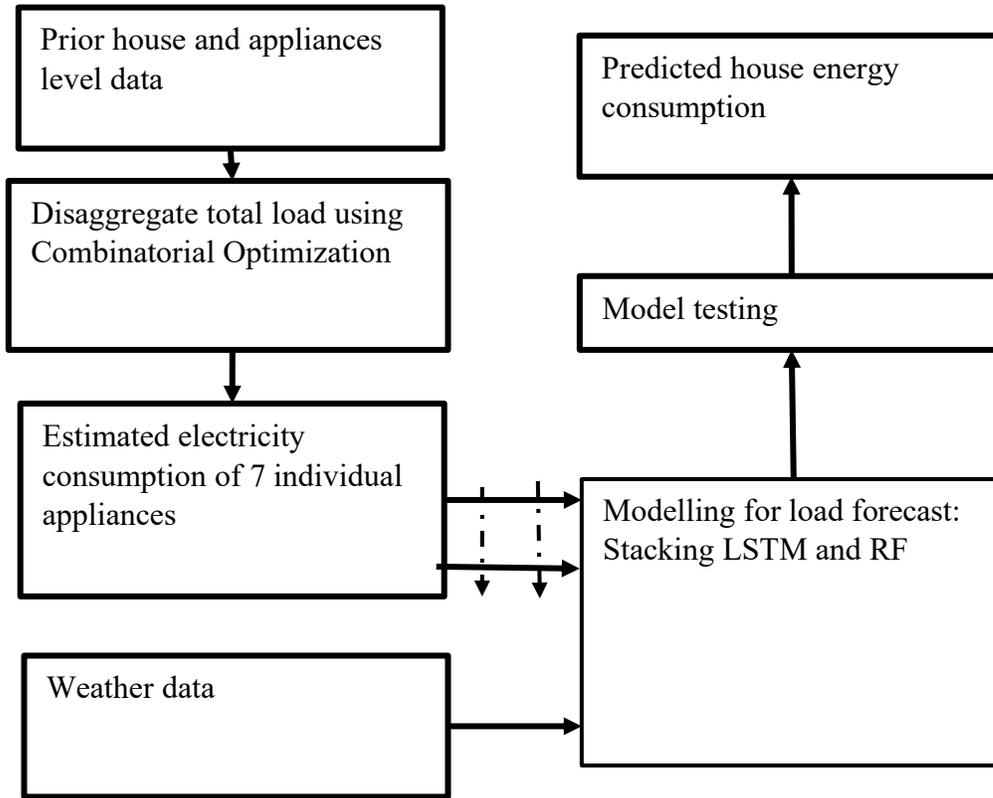


Figure 6.1: Overall conceptual model of proposed household load forecast based on load disaggregation

### 6.1. Combinatorial optimization-based load disaggregation

The architecture and training process of load disaggregation is shown in fig 6.2. The general appliances modeling started by data visualization in which the appliance's profile and states were obtained using NILMTK functionalities [14]. The combinatorial optimization (CO) approach was used to separate the appliances' energy consumption from the aggregated smart -meter power demand.

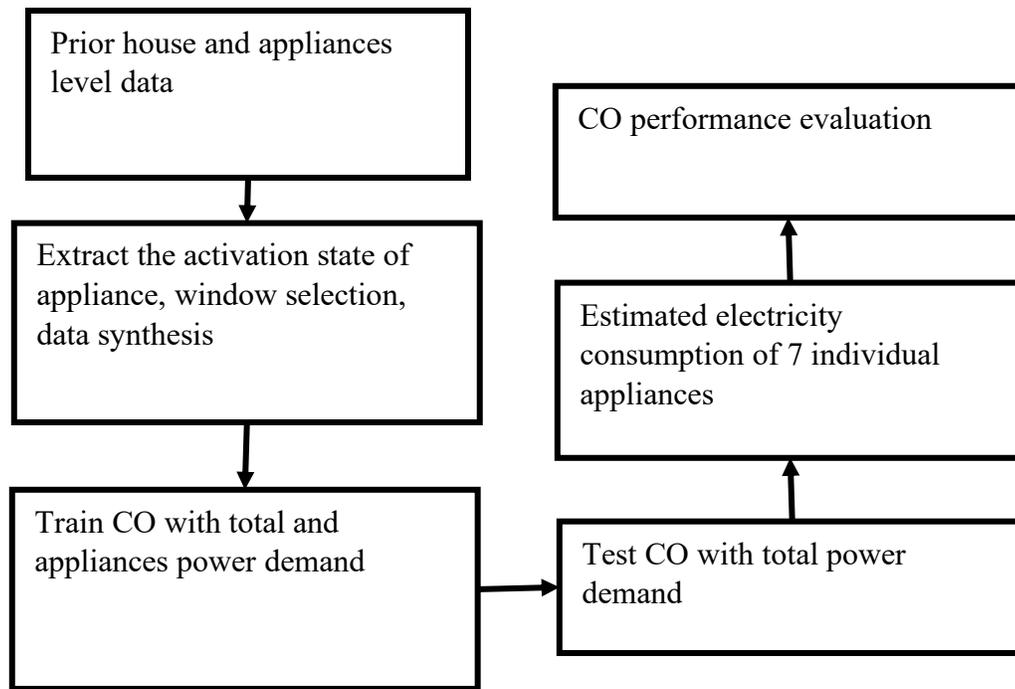


Figure 6.2: Steps for appliances power estimation based on Combinatorial optimization

### 6.1.1 Data Preprocessing

We used the Almanac of minutely power dataset (AMPDS) [21] that contains the appliances energy consumption and aggregated power demand of a single house in Canada, and that was recorded every minute for 2 years. We decided to include the eight most-used classical appliances in the house and the choice of the appliances was motivated by the fact that we wanted to investigate the ability of the CO approach to disaggregate the appliances that consume the large portion of home power demand.

The appliances that were targeted in the load disaggregation process are Heat pump, air furnace fan and thermostat, the cloth dryer, light, fridge, Television, and sockets. The chosen 7 appliances are the most used and constitute the high energy proportion of the total house energy consumption. As described in the literature [4], the fridge falls in the finite

and multi-state types of appliances whereas the microwave, sockets, TV, cloth dryer are classified into the finite or continuously varying signal type of appliances. Having appliances of the same states helped to evaluate the ability of CO to distinguish the appliances of similar patterns (i.e. power wave flows).

Features included in the disaggregation model are defined as follows [21]:

1. Heat pump: The actual energy consumption of heat pump in the house 1, it is was measured in Watt and is denoted heat pump.
2. Cloth dryer: The actual energy consumption of cloth dryer in house 1, is was measured in Watt and is denoted cloth washer.
3. Fridge: The actual energy consumption of the fridge in the house 1, it is was measured in Watt and is denoted Fridge.
4. Air furnace fan and Thermostat: The actual energy consumption of air furnace fan and thermostat in the house 1, it is was measured in Watt and is denoted air furnace fan and thermostat.
5. Lighting: The actual energy consumption of light in the basement of house 1, is was measured in Watt and is denoted lighting.
6. Television: The actual energy consumption of TV in house 1, it is was measured in Watt and is denoted Television.
7. Sockets: The actual energy consumption of sockets and plugs in the house 1, it is was measured in Watt and is denoted Sockets.

8. Data synthesis(aggregated): the synthetic data were obtained by adding the power demand of all 7 appliances under consideration. It is denoted Site meter

For each of the 7 appliances, the following steps have been used to determine the state of activation and window size of a specific appliance [36].

A. Activation state of appliance: With data, visualization using the get activation function available in the NILMTK python module, the active and inactive state of each appliance has been identified as shown in Figure 1.1.

B. Window selection. This was intended to define the low and up signal limit for the ON state of the appliance. This helped in filtering out and remove outliers in the input signals.

### **6.1.2. Detailed technical design of Combinatorial optimization for load disaggregation**

We designed a combinatorial optimization algorithm for disaggregating appliance energy consumption by assuming that: (1) The total energy consumption in the house is equal to the sum of 7 individual appliances under consideration. (2) At a given time, the appliance can only be in a single state (e.g. OFF/ON). (3) We can only have one appliance type (e.g. heat pump) in the target house. The challenge is that it is possible to have different appliances with the same state type, therefore, the design of the Co was done so that it able to separate the appliances with the same state in the house.

The individual appliance consumption was obtained by solving the energy disaggregation problem based on similar consumption patterns.

The objective function is defined as the minimization of the error  $\hat{e}(t)$  between the aggregate and sum of appliances power demand and is formulated as follows [40,50].

$$\hat{e}(t) = \min \left| \hat{y}(t) - \sum_{i=1}^N a(t) \cdot y^{(i)}(t) \right| \quad (6. 1)$$

Where  $\hat{y}(t)$  is the aggregated household power demand,  $y^{(i)}(t)$  is the power demand of appliance  $i$ , and  $N$  is the number of appliances that are present in the house.  $a(t)$  represents the state of the appliance at a time  $t$ . For the ON/OFF state of the appliance, this factor becomes 1 and it becomes 0 for all other scenarios.

The choice of combinatorial optimization-based pattern similarities is aimed to filter out the similar reading samples and eliminate the irrelevant energy reading samples in energy consumption of appliances to achieve data unification and reduce the complexity of computation.

The training set is defined as: (1) Training set that contains the appliances level data of 8 individual appliances in house 1 of AMPDs for the same sampling frequency and length as the total power to be disaggregated. (2) The training set  $z$  is made with the synthetic data (total power) which is in the function of the energy consumption of 7 important appliances that are present in a single house of AMPDs. The entire training set can be defined as [51]:

$$z_t = f(w_t^1, w_t^2, w_t^2, w_t^3, \dots, w_t^m) \quad (6. 2)$$

Where  $m$  represent the number (i.e.7) appliances included in the CO model training. Those input data were trained to obtain output vector  $y$  representing the appliance signal  $y^{(i)}(t)$ .

The vector that contains the estimated configuration of appliance  $y^{(i)}(t)$  is obtained from the aggregated power consumption  $\hat{y}(t)(t - d)$  with the following equations.

$$(y^{(1)}(t), y^{(2)}(t), \dots, y^{(n)}(t)) := w, z(\hat{y}^{(i)}(t - d), \dots, \hat{y}^{(i)}(t), \dots, \hat{y}^{(i)}(t + d)) \quad (6.3)$$

where  $d$  represent the interval or delay to the neighborhood samples and  $t$  represents the continuous time.

The number of slices is minimized so to achieve the estimated appliance signal  $\hat{y}^{(n)}$  that is closer enough to the measured appliance signal  $y^{(i)}(t)$ .

The Improved CO algorithm works in an iterative schema by checking the similarity between appliance samples as indicated in table 6.1. In the training stage, the algorithm is intended to train the training set  $z_t$  and build an array  $A_z$  of training samples that will be used in the testing stage.

The algorithm works as follows: In the training stage, the algorithm starts by iterating on each training sample  $z_i$  and keeps tracking the neighbor sample  $z_j$ . If a neighboring sample has a similar energy consumption with the current sample, the iteration is increased. The array  $A_z$  that contains the trained sample values is stored to be used in the testing stage for clustering them with other samples whose consumption is like that processed sample.

In the testing stage, the algorithm builds an array  $A_y$  that contains the signature feature of each processed testing sample  $y_i$ . In the same way, as in the training stage, the counter is increased if the current testing sample contains a sample with the same energy as the neighboring sample  $y_i$ . The testing stage also contains the iterative loop that checks the similarities between each testing sample with training samples. It compares each training

sample  $z_j$  with the current testing of processed consumption  $y_i$ . As shown in line 14-30, If the difference between the two samples is less than the set threshold and the consumption of a testing sample is greater than the minimum power consumption  $H$ , the sample is added in the set that will be used for the next comparison.

At the final stage, the algorithm utilizes the trained samples that are stored in the set  $D$ . If the set  $D$  contains at least one trained sample, then samples that minimize the difference between signature features ( the processed training  $A_z$  and testing sample  $A_y$  ) are selected and one of them is randomly chosen as the output prediction as shown inline 30-34. If non-trained similar samples are found in set  $D$ , then, the original training samples that minimize the difference between the consumption are used the output prediction as shown in lines 35-37. In the end, the output predictions are mapped with the actual appliance energy consumption [40,50].

Table 6.1: Customized Pseudocode of customized CO based patterns similarity, adapted from [50].

A. CO Function for Training the appliance level data: Array of set $A_z$ )
<pre> 1. <math>A_z \leftarrow</math> Training array of length <math>Z</math> 2. <math>counter(i) \leftarrow 0</math> 3. <math>j \leftarrow 0</math> 4. Take sample <math>z_j</math> from the training array <math>A_z</math> 5. for all <math>\{z_j \in Z:  j - i  &lt; d\}</math> do # <math>d =</math> sampling frequency 6.     if <math>z_j &gt; z_i - \delta</math> then # <math>\delta =</math> threshold 7.     <math>counter(j) \leftarrow j + 1</math> 8.     end if 9. end for 10. if <math>i =</math> maximum iteration or length <math>A_z = 0</math>, then 11. break loop iter 12. end if 13. <math>counter(i) \leftarrow i + 1</math> </pre>
B. CO Function for testing and disaggregate the total house data: Array of testing set $A_y$
<pre> 14. <math>A_x \leftarrow</math> Testing array of length <math>Y</math> 15. <math>counter(i) \leftarrow 0</math> 16. <math>j \leftarrow 0</math> 17. Take sample <math>z_j</math> from the testing array <math>A_y</math> </pre>

```

18. for all  $\{y_j \in Y: |j - i| < d\}$  do
19.     if  $y_j > y_i - \delta$  then #  $\delta = \text{threshold}$ 
20.          $\text{counter}(j) \leftarrow j + 1$ 
21.     end if
22. end for
23. if  $i = \text{maximum iteration or length } A_z = 0$ , then
24. break loop iter
25. end if
26.  $\text{counter}(i) \leftarrow i + 1$ 

```

C. CO function for disaggregating: combine similar samples (Array  $A_z$  of trained appliance data and array  $A_y$  of trained house total data.

```

27. for all  $y_j \in Y$  do # disaggregate total power
28.  $D \leftarrow \emptyset$ 
29. for all  $z_j \in Z$  do. # check similarities
30.     if  $|z_j - y_i| \leq \delta$  and  $y_i > H$  then. # H is the low power consumption
31.          $D \leftarrow D \cup \{j\}$  # store samples
32.     end if
33.     if  $|D| \geq 1$  then. #get samples from store
34.          $j \leftarrow \arg \min\{|A_z(D(\cdot)) - A_y(D(\cdot))|\}$  # map samples that minimize the difference
35.     else
36.          $j \leftarrow \arg \min\{|z(\cdot) - y(\cdot)|\}$  # map other samples
37.     end if
38.      $k \leftarrow \text{rand}\{1, \dots, \text{length}(D)\}$ 
39.      $y(i, \cdot) \leftarrow w(I(j(k), \cdot))$ 
40. end for

```

The training process of the CO approach for this experiment is described as follows: The REDD dataset was first converted into the hierarchical data format (HDF5) to analyze data using the NILMTK toolkit. The CO model was trained using sub-metered appliance information and the trained model was used to disaggregate the aggregated electricity consumption to obtain the state and power consumption of each appliance present in that building.

The AMPDs dataset was resampled to the frequency of 1 min. The training window was set to be in the range between 2012-01 -01 to 2012-10-30 whereas the testing window has been chosen to be from 2012-11-01 to 2013 -04-30. Appliances with different states were included in the training set to evaluate the efficacy of the CO-based similar patterns algorithm. Different metrics were used to evaluate the disaggregation performance of

combinatorial optimization. The implementation of this experiment of the CO based pattern similarity for disaggregating the signal features of each appliance from the aggregated house energy consumption was adapted from [14,50].

The choice of combinatorial optimization (CO) based patterns similarity approach was motivated by the fact that the experiment results discussed in chapter 5 revealed that the CO-based approaches provide the improved disaggregation score when the number of appliances in the model is increased [6,36]. Moreover, utilizing the patterns similarities-based approach helped in separating the appliances with the same states such as heat pump, and air furnace fan.

## **6.2. Load forecasting based on load disaggregation**

The objective of the project is to combine the estimated power consumption of individual appliances with weather data to forecast household power demand. The training set consists of the disaggregated energy consumption of 7 classical appliances located in one house in Canada and hourly weather data information obtained from the environment Canada's YVR weather station (Vancouver international airport).

As shown in figure 6.3, we did data visualization and provide energy individualized energy statistics for each of the 7 appliances. This helped to access the appliances' energy use and recognize the appliances that consume the great portion of home power demand. The appliances' data were resampled to an hourly basis to match the weather. We did re-sampling to construct the new periodic samples from the original samples. We used a linear interpolation method between neighboring samples. The data normalization was done to make the model function to apply the appropriate weights to all the features. The values of

the features were downscaled between 0 and 1 using Min-max scaler functions available in python. The ensemble learning that combines predictions of random forest (RF) and long- short term memory (LSTM) network is used to enhance the short-term forecasts for a small house with high uncertainty. The choice of LSTM-RF network was motivated by the fact the LSTM network has adjustable internal cell memory and can understand the nonlinearity among weather inputs whereas the random forest model can easily understand the time series dataset.

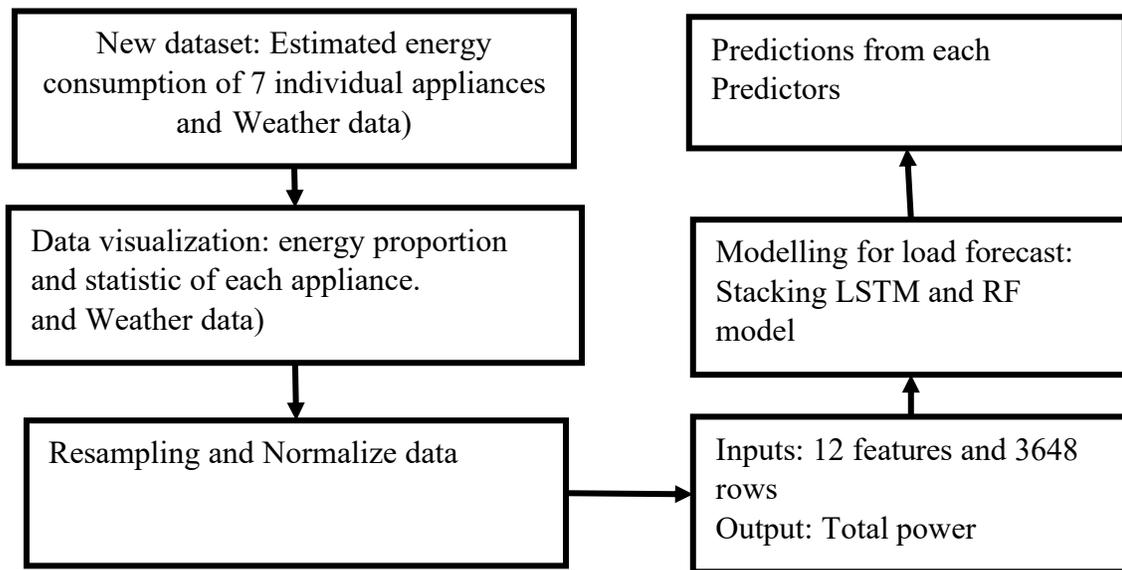


Figure 6.3:Process of heterogeneous ensemble learning based load forecasting

### 6.2.1. Data Description

The proposed load forecast used the disaggregated appliances energy consumption and weather data with the following labels:

1. Heat pump ( $P_H$ ): The estimated electricity consumption of the heat pump. It is in Watt and is denoted  $P_H$ .

2. Cloth washer ( $P_C$ ): The estimated electricity consumption of a cloth dryer. It is in Watt and is denoted  $P_C$ .
3. Air furnace fan and Thermostat ( $P_A$ ): The estimated electricity consumption of an air furnace fan and thermostat. It is in Watt and is denoted  $P_A$ .
4. Lighting ( $P_L$ ): The estimated electricity consumption of plugs and lights. It is in Watt and is denoted  $P_L$ .
5. Fridge ( $P_F$ ): The estimated electricity consumption of plugs and lights. It is in Watt and is denoted  $P_F$ .
6. Sockets ( $P_S$ ): (The estimated electricity consumption of plugs and lights. It is in Watt and is denoted  $P_S$ .
7. Television ( $P_T$ ): (The estimated electricity consumption of plugs and lights. It is in Watt and is denoted  $P_T$ .
8. Temp ( $T$ ): The temperature at YVR weather station. It is measured in Celsius and is denoted  $T$ .
9. Stn Pressure ( $P$ ): The standard pressure at YVR weather station. It is measured in kPa and is denoted  $P$ .
10. Wind spd ( $W_S$ ): The wind speed at YVR weather station. It is measured in Km/h and is denoted  $W_S$ .
11. Rel Hum ( $R_h$ ): The relative humidity at YVR weather station. It is measured in % and is denoted  $R_h$ .

12. Visibility(V): The visibility at YVR weather station. It is measured in Km and is denoted  $V$

The estimated appliances -level data were resampled to one hour to match with weather data. This provided the data with 12 features and 3648 observations that were used to predict the house power demand. We selected 7 classical appliances to affect more the whole house power demand and the energy consumptions of lighting and heat pump are related to the environmental conditions.

The training set at instant time  $t$  can be expressed with the following vector  $y(t)$  which combines the vector contribution of 12 inputs:

$$x(t) = (P_H(t), P_C(t), P_A(t), P_L(t), P_S(t), P_T(t), P_F(t), T(t), P(t), W_S(t), R_h(t), V(t))$$

where  $P_H(t)$  is the estimated power demand for Heat pump;  $P_C(t)$  is the estimated power demand for Cloth dryer;  $P_A(t)$  is the estimated power demand for air furnace fan and thermostat;  $P_L(t)$  is the estimated power demand for lighting ;  $P_A(t)$  is the estimated power demand of fridge,  $P_A(t)$  is the estimated power demand of sockets;  $P_A(t)$  is the estimated power demand of television at time  $t$ ;  $T(t)$  is the outside temperature;  $W_S(t)$  is for the wind speed ;  $R_h(t)$  is for relative humidity and  $V(t)$  is the outside visibility at time  $t$ .

### **6.2.2. Stacking Modelling for Load Forecasting**

The stacking model is a kind of ensemble learning framework in which predictions from multiple individual predictors are integrated to obtain the final predictor. The stacking is conducted in two stages, where in the first stages, the individual predictors are trained with the original data to obtain their output predictions. In the second stage, the output

predictions from individual learners are combined to form the second level data that is used to train the final model and obtain the integrated output.

After data exploration and feature engineering, the data were normalized and divided into training and testing sets. As shown in figure 6.4, The training set was made with 3648 observations and 12 features that cover 4 months data (2012-11-01 to 2013-03-01) and it was inputted to both random forest regressors (RF) and Long short term memory(LSTM) models. The results from the two predictors (RF and LSTM) were used as the input to the final linear regression (LR) model to predict the final house energy consumption.

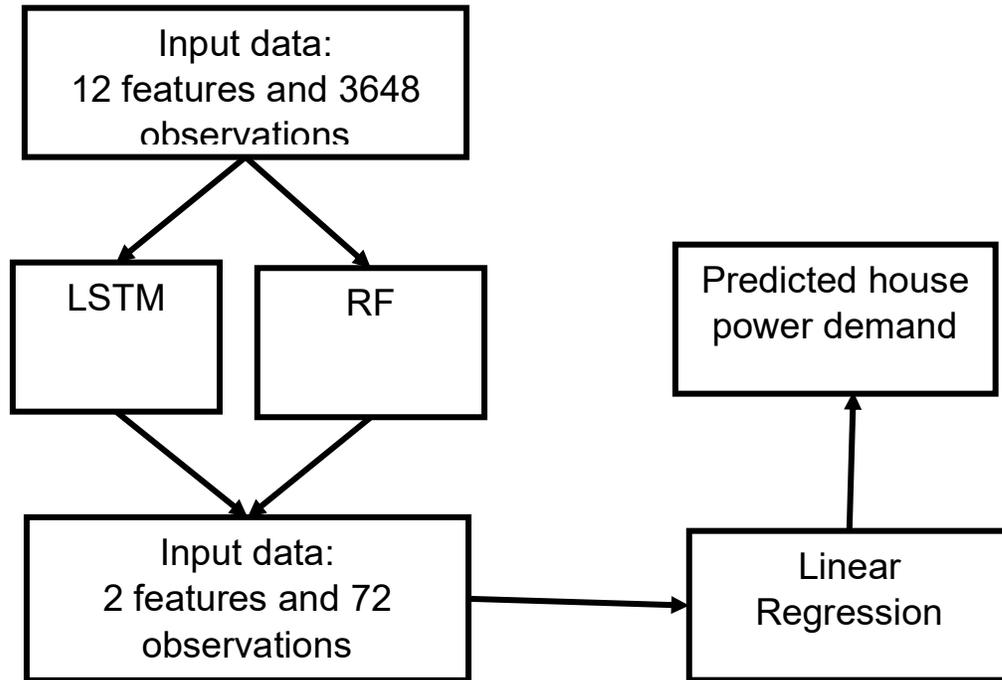


Figure 6.4:Stacking load forecasting model

#### A) Random Forest regressor architecture for load forecast

The random forest technique uses the decision tree algorithm that constructs the target or predictions based on many observations. In ensemble learning, the generated number of classifiers are summed up to obtain the result [42]. We experimented by tuning the number

of trees and depth of trees with the brute force method. The number of trees was in the 1-500 range whereas the depth trees were in the 1-10 range.

The model training was conducted with data features that cover 4 months (2012-11-01 to 2013-03-01) in the dataset [8]. After gradient boosting model training, the hyperparameter tuning based on cross-validation via Grid search was utilized to loop over and obtain the optimal parameter from range (number of trees=500, the height of trees=10). The Model testing has been done using the aggregate power demand data of 1-month data (2013-04-01 to 2013-04-30). As shown in Table 6.3, for the random forest, we used the grid search of python that loop up from the search space and select the parameter that helps the model to achieve the low forecasting error.

The random forest training process involves the following steps:

**Step 1:** Re-sampling the estimated energy consumption of appliances to match the hourly weather data.

**Step 2:** Normalize the time series inputs using scikit-learn preprocessing to scale the original inputs from 0 to 1.

**Step 3:** split data into training and testing range and input the training data to the random forest model

**Step 4:** Perform the hyperparameter tuning based on cross-validation via Grid search to obtain the optimized parameters.

**Step 5:** Test the model performance with the house total power demand and obtain predictions

**Step 6:** Save predictions and calculate the forecasting performance of a regression-based model.

Table 6.2: The optimal parameters and search spaces of forecasting algorithms.

Model	Parameters	Used Value	Search space
LSTM	Number of layers	3	Between 1 to 10
	Number of hidden neurons in layer 1	612	Between 1 to 2016
	Number of hidden neurons in layer 2	306	Between 1 to 2016
	Number of hidden neurons in layer 2	156	Between 1 to 2016
	Input shape	1,12	-
	Dropout	0.2	Between 0.04 to 0.80
	Dense layer	1	-
	Learning rate	0.0001	[0.0001,0.005,0.001]
	Training epochs	25	Between 20 to 50
optimizer	Adam	[sgd, Adam]	
Random Forest	Number of trees	500	Between 50 to 5000
	height	10	Between 1 to 30
Linear Regression	Meta learner	-	-

### B) Long-Short Term Memory (LSTM) for load forecast

The recurrent neural network is a type of network that can memorize the previous information and use that information to learn the current data. It is found to be suitable for solving the problem that involves the sequence of data such as the time series home energy consumption. In the recurrent network, the decision at the time is based on the decisions made from the previous time step  $t - 1$ . The long short-term memory (LSTM) is a kind of recurrent network that can keep its memory state during the sequence cycle. It is made with the input gate  $i_t$  that maintains its inner state; the forget gate  $f_t$  which helps in filtering the previous inputs; the output gate  $o_t$  for deciding the data to feed in the LSTM internal state[48,49]. Based on the input signal  $g_t$ , the LSTM internal cell state  $s_{t-1}$  communicates with the inputs  $x(t)$  and previous output  $h_{t-1}$  to determine what inputs to maintain, update,

and delete. At the instant time  $t$ , the state of LSTM internal cell can be expressed as follows:

$$\begin{aligned}
 f_t &= \sigma(W_{fx}x(t) + W_{fh}h_{t-1} + b_f) \\
 i_t &= \sigma(W_{ix}x(t) + W_{ih}h_{t-1} + b_i) \\
 g_t &= \phi(W_{gx}x(t) + W_{gh}h_{t-1} + b_g) \\
 o_t &= \sigma(W_{ox}x(t) + W_{oh}h_{t-1} + b_o) \\
 s_t &= g_t \odot i_t \odot s_{t-1} \odot f_t
 \end{aligned} \tag{6.4}$$

Where  $w$  is the weight for specific input to the gate,  $\sigma$  is the activation,  $\phi$  is for the loss function and  $\odot$  illustrates element wise multiplication.

The LSTM training process involves the following steps:

**Step 1:** Re-sampling the appliances power demand data to obtain the hourly power data that match with the weather.

**Step 2:** Normalize the time series data set using scikit-learn preprocessing to scale the original data from 0 to 1.

**Step 3:** split data into training and testing range and input the training data to the gradient boosting model. Reshape into a 3-dimensional array (samples, timesteps, and features) using a NumPy array of python preprocessing.

**Step 4:** Define the architecture of the LSTM network and loss function. Optimize the model and add dropout to avoid the model overfitting.

**Step 5:** Test the model performance with the total household power demand and obtain predictions

**Step 6:** Save predictions and calculate the forecasting performance of an LSTM based model.

The hyperparameter selection process includes selecting the number of layers in the LSTM network, the number of hidden neurons in each LSTM layer. The grid search and manual search are the two most used strategies for determining the parameters that optimize model performance.

As shown in Table 6.2, for the LSTM network, we used the manual search that selected the hyperparameters from the search space, used them to train the model, and calculate the forecasting error. The hyperparameter search is repeated many times until the search space is completed, and we select the parameter that helps the model to low forecasting error.

As shown in figure 6.5, the LSTM network was designed to have one input layer with its length is determined by the appliance duration; three LSTM layers stacked together and one output fully connected layer with linear activation. With dynamic window size (from 1 to 2016), the optimal parameters (i.e. neurons) were obtained for LSTM layers. The number of neurons for the first LSTM layer is 612, the second LSTM layer has 306 neurons and 156 neurons for the third LSTM layer.

The linear activation was used to transform the output obtained from each unit layer as the input to the next layer. A dropout of 20% was added to avoid the model overfitting during training. The mean square error (MSE) loss function was selected whereas Adam optimizer with a learning rate of 0.0001. The number of training epochs was 25, and a batch size of 72 was used.

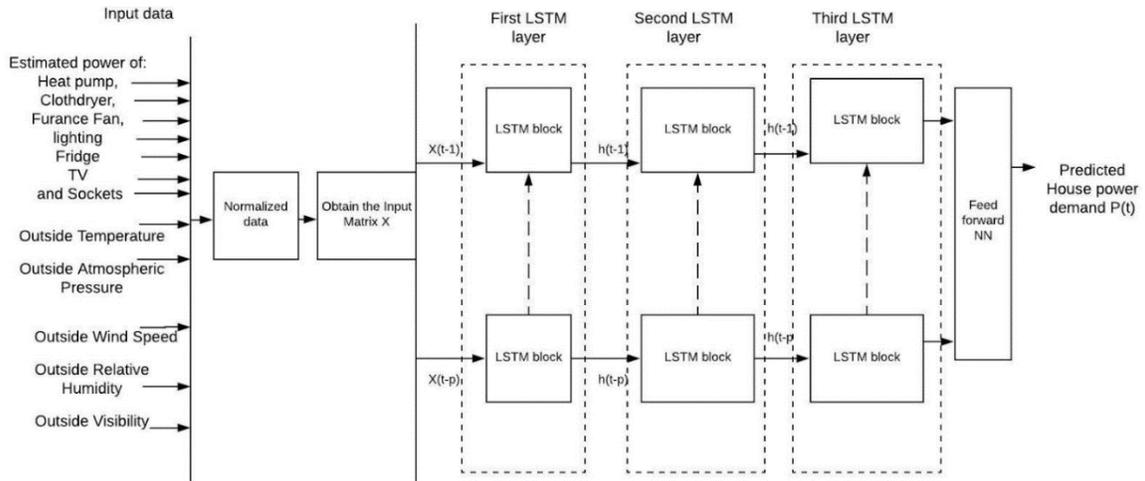


Figure 6.5: Customized LSTM architecture with appliances power demand and weather data, adapted from [48].

### C) Stacking model based on LSTM and RF predictions

The stacking model combines predictions from two individual predictors into one final forecasting system. We used the predicted results of two predictive models (RF and LSTM) as the input set in the final model. It was assumed that the final model (linear regression) predicts the house load for the same forecasting period as the two individual predictors.

The stacking model can be explained in the following stages:

**STEP 1:** Each predictor(model) is trained with the training set  $x(t)$  and is tested with the total power  $P(t)$  to achieve the predicting power demand  $P_T^M$  where M is the number of predicting models) and T is the forecasting period. In this project, we used two single models' predictions, and this resulted in two load predictions results

$$P_T^1 = (P_1^1, P_2^1, \dots, P_T^1) \text{ for RF model and } P_T^2 = (P_1^2, P_2^2, \dots, P_T^2) \text{ for the LSTM model.}$$

**STEP 2:** The predictions from the two single models were combined with actual total household power demand to obtain the input to the final linear regression model to obtain the final predictions load vector  $P(t)$  expressed by the following expression[26,50].

$$P(t) = \sum_{M=1}^2 w_T P_T^M$$

Where  $w_T$  represents the weight of the linear combination at the forecasting period  $T$  and is obtained through the meta-learning process.

Table 6.3: Customized pseudocode of RF-LSTM Ensemble for load forecasting, adapted from [26].

```

Input data:  $x(t) = \{(x_1, y_1), (x_1, y_1), \dots, (x_T, y_T)\}$ 
Testing data:  $x_i = (x_i, y_i)$  # Actual total power with  $i > T$ 
Output:  $P(t)$ 
  Learning model  $M_i$  with  $i = 1, 2$  and  $M_f$  #  $i$  is the number of predictors to be trained
  with first level data and  $M_f$  is the final linear model to be trained with second
  level data.
  Implementation:
Training stage1: the SVR and LSTM model using the original data  $x(t)$ 
                     $x(t) = \{(x_1, y_1), (x_1, y_1), \dots, (x_T, y_T)\}$ 
  for  $i = 1, \dots, p < T$ :
     $M_T(t) = M_i x(t)$  # Train RF and LSTM with original data
  end for,
Testing Stage: Obtain predictions with Trained model  $M_T$ 
  for  $t = 1, \dots, T$  do
    for  $t = 1, \dots, T$  do
       $M_{P_i}(t) = M_T x(i)$  # use the trained mode to get model predictions  $M_{P_i}$ 
    end for
  end for
  The new dataset  $x'(t) = M_{P_i}(t) \cup (x_i, y_i)$  # combined RF and LSTM predictions with actual
  testing data
Training stage2: Train the linear regression using the new dataset  $x'(t)$ 
  for  $t = 1, \dots, T$  do
     $M_{P_f}(t) = M_f x'(t)$ 
  end for
 $P(t) = M_{P_f}(M_{P_1} x(1)), \dots, (M_{P_T} x(t))$  # output

```

## CHAPTER 7: EVALUATION AND RESULTS

In this chapter, we present the results of the load forecasting based on load disaggregation. Section 7.1 presents the functional results obtained using different evaluation metrics. Section 7.2 discusses efficient evaluation. Section 7.3 discusses the complexity of the proposed load forecasting. Section 7.4 presents a comparison of the results.

### 7.1 Functional evaluation

The data extraction has been conducted using the Almanac of minutely power dataset (AMPDS)[14] that contains the appliances and house level energy data of a single house in Canada that were recorded every minute for a duration of 2 years. The extracted energy consumption of 7 appliances and weather data have been used to evaluate the proposed LSTM-RF ensemble learning-based load forecast approach.

$y^{(i)}(t)$  = the actual energy consumption of an appliance  $i$  at the instant time  $t$

$\hat{y}^{(i)}(t)$  = the estimated energy consumption of an appliance  $i$  at the instant time  $t$

Taking the example of the lighting load, the following metrics can be defined [3,4]:

*MAE* : The mean absolute error (MAE) indicates the relative error in the energy attributed to every appliance at a time  $t$ .

*RMSE*: The root mean square error metric helps in evaluating the scattering score of the forecasting model.

*NRMSE*: The normalized root mean square error allows performance comparison for the model tested with different appliances dataset with different specifications.

*MAPE*: Mean absolute percentage error.

$$MAE = \frac{\sum_{t=1}^T |\hat{y}^{(i)}(t) - y^{(i)}(t)|}{T} \quad (7.1)$$

$$MAPE = \frac{1}{T} \frac{\sum_{t=1}^T |\hat{y}^{(i)}(t) - y^{(i)}(t)|}{y^{(i)}} \times 100 \quad (7.2)$$

$$MAE = \frac{\sum_{t=1}^T |\hat{y}^{(i)}(t) - y^{(i)}(t)|}{T} \quad (7.3)$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^T (\hat{y}^{(i)}(t) - y^{(i)}(t))^2}{T}} \quad (7.4)$$

$$NRMSE = \frac{RMSE}{y_{max} + y_{min}} \quad (7.5)$$

Before starting the load disaggregation, we did data visualization and get the energy proportion of 7 important appliances in the house. As shown in figure 7.1, the heat pump and air furnace fan and thermostat account more than 50% of the aggregated energy consumption of 7 appliances whereas the sockets consume the least mean power in the same house.

The combinatorial optimization-based pattern similarity model was trained with energy signal from 7 appliances to separate the aggregated power demand (i.e. site meter) into the individual appliance energy consumption. As shown in Figures 7.2 and 7.3, the energy proportion of the consumption of the estimated appliance is in line with the actual energy proportion in figure 7.1. Disaggregated energy consumption of heat pump and air furnace fan and thermostat cover over 50% of the aggregated power demand, followed by the cloth dryer (11.5%) and fridge (10.9%). The high energy consumption for heat pump and air furnace fan and thermostat could be originated from the fact that they are found in ON state for most of the daytime to adjust the temperature inside the house.

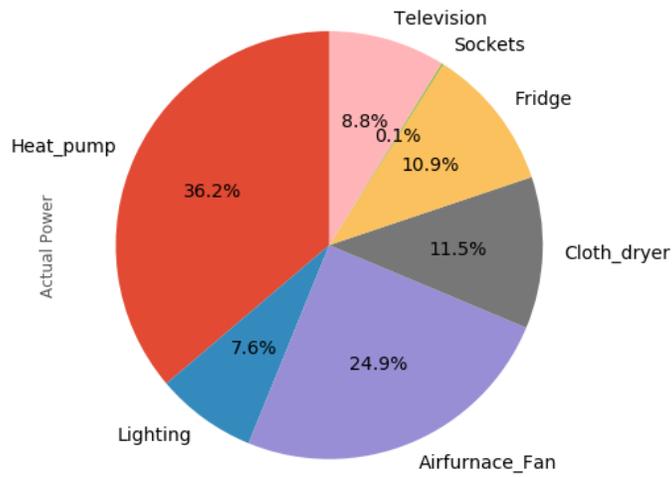


Figure 7.1: Proportion of appliance energy in the actual power demand from AMPDs dataset.

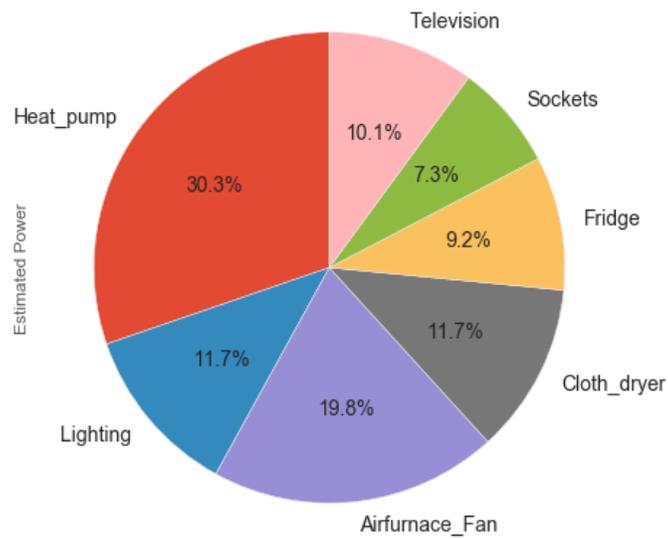


Figure 7.2: Proportion of appliance energy in the CO- predicted power demand from AMPDs dataset.

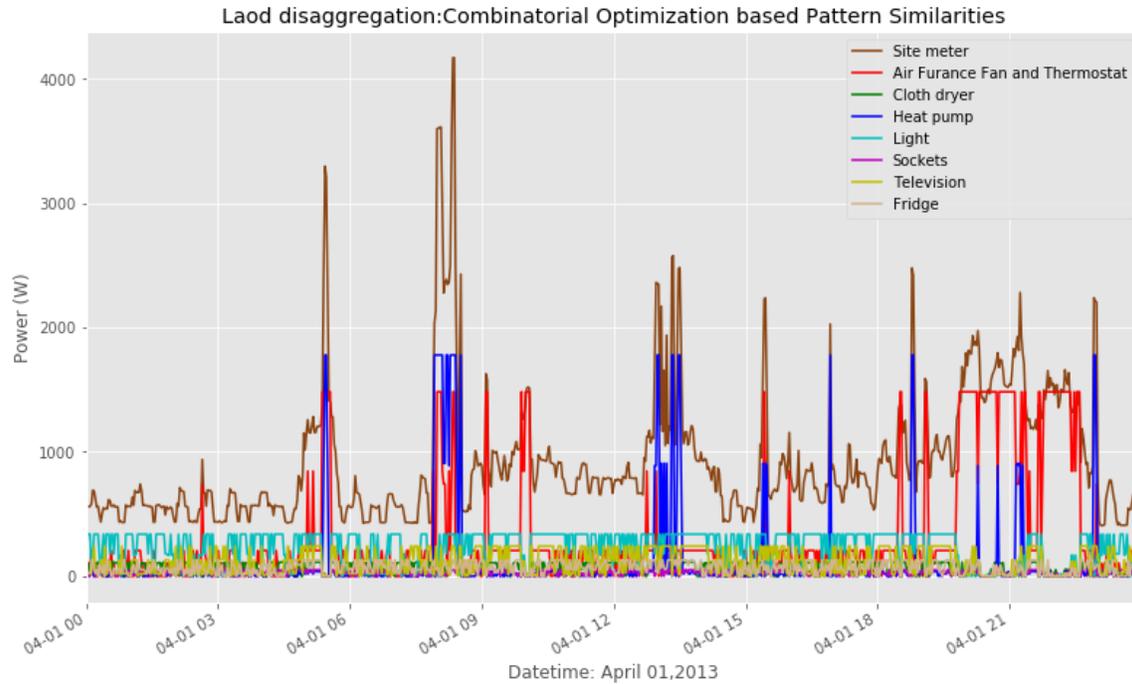


Figure 7.3: The disaggregated energy consumption of appliances with CO on AMPDs

In terms of load disaggregation performance, the results in figure 7.4 and table 7.1 illustrate that in terms of F1 score, the CO succeeded (over 0.8) to disaggregate the heat pump, lighting, air furnace fan, sockets, and television. The least disaggregated F1 scores were obtained for the fridge (0.43) and cloth dryer (0.52). This

indicated that the CO approach is more capable of separating the ON/OFF type of appliance as opposed to the always ON (fridge) type of appliance.

Table 7.1:CO energy disaggregation performance in home 1 seen during AMPDs data training.

CO disaggregation performance			
Appliance type	Recall	Precision	F1 score
Heat Pump	0.94	0.93	0.94
Lighting	1	0.99	0.99
Fridge	0.40	0.46	0.43
TV	1	0.99	0.99
Sockets	1	0.78	0.87
Air furnace fan	0.99	1	0.99
Cloth dryer	0.49	0.58	0.52

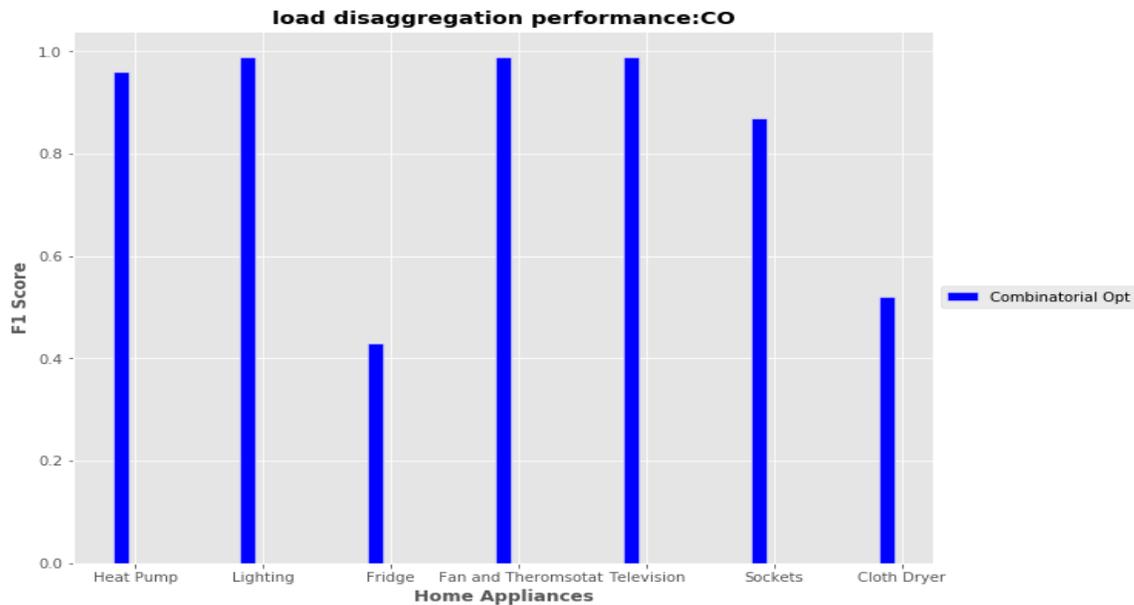


Figure 7.4:CO disaggregation F1 score for different appliances present in home 1 seen during AMPDs data training.

The obtained estimated appliances energy consumption was combined with the environment weather data to evaluate the LSTM-RF ensemble learning-based load forecasting approach. Figure 7.5 illustrates the actual and predicted household total power demand by different methods. As can be seen, the predicted total power demand agrees

with total power measured by the site meter for the 3 days ahead. The choice of 3 days ahead forecast was motivated with the fact that short term forecasts (i.e. less than a week) are more accurate than long term forecasts (i.e. over a week) and home residents need an accurate forecast to take decide about cutting or reduce the house energy consumption. In terms of mean energy consumption, the predicted energy consumption was 1331Watts compare to 1335 Watts of the actual house energy consumption.

As shown in table 7.2 and figure 7.5, the performance comparison indicates that the proposed load forecasting based on load disaggregation achieved over 96% in terms of accuracy. As presented in figure 7.5, the ensemble learning (CO+LSTM+RF) outperforms the single method (CO+LSTM) and CO+RF. In terms of normalized root mean square error (NRMSE), the ensemble learning reduced the error up to 0.0411 compared to 0.0706 of long short-term memory (LSTM) and 0.0423 of Random forecast (RF). This could be credited to the fact that the ensemble learning exploited the ability of LTSM based deep learning to understand the non-linearity among weather information.

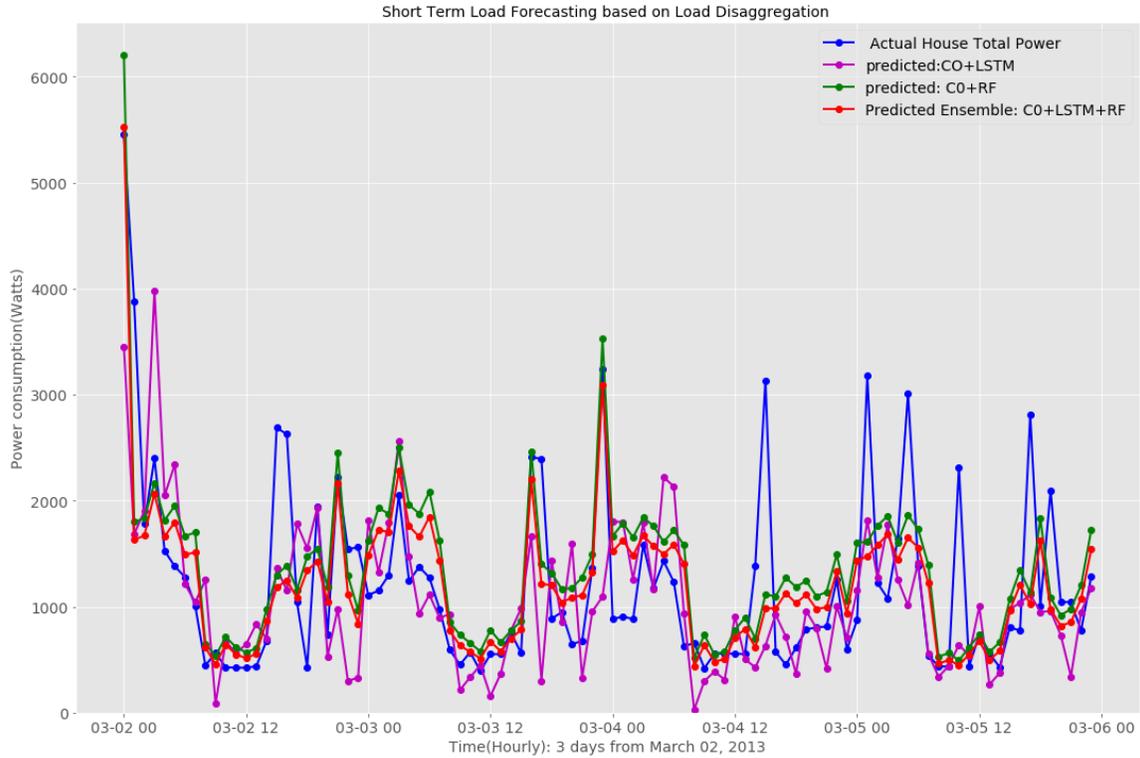


Figure 7.5: The actual house power demand and predicted house total power demand by different forecasting models.

Table 7.2: Performance comparison of load forecasting models based on load disaggregation.

<b>CO+Ensemble model results for 3 days: with both disaggregated energy consumption of appliances and weather data</b>					
Method	Inputs	Error Metrics			
		RMSE	NRMS E	MAE	Accuracy%
RF	Disaggregated energy consumption of 7 appliances and weather data	0.0377	0.0423	0.0160	96.23
LSTM		0.0739	0.0706	0.0513	92.60

Ensemble	RF+LSTM predictions	0.0386	0.0411	0.012	96.13
----------	---------------------	--------	--------	-------	-------

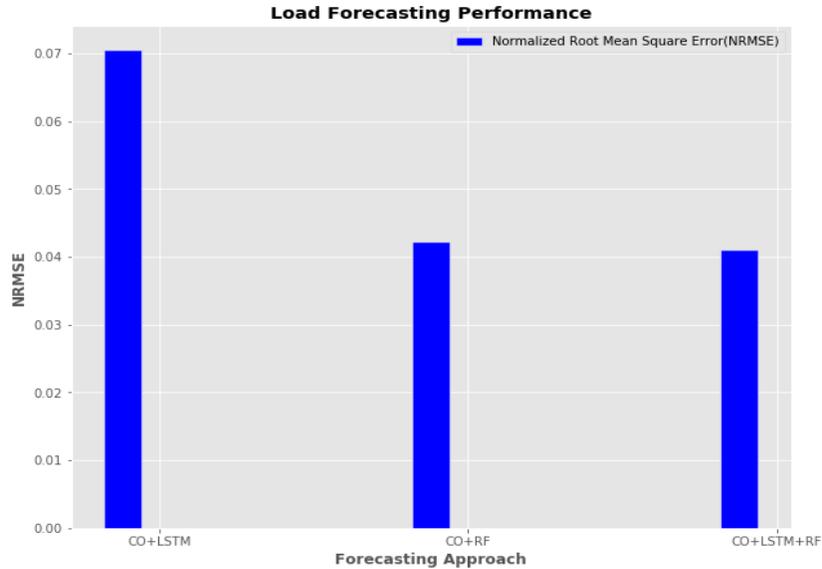


Figure 7.6: NRMSE comparison of load forecasting models based on load disaggregation

## 7.2 Efficiency evaluation

The efficiency is defined as the capability of the proposed method to give disaggregation and forecasting output in real-time. The effectiveness of the proposed load forecasting based on load disaggregation was conducted to evaluate if proposed load forecast architecture can locally be implemented on the personal computer and minimize the privacy concerns of energy consumers in the case of cloud computing.

The computational time covers the training and disaggregation time of CO and training of LSTM-RF ensemble learning-based load forecasting. The CO was trained using real-household data with 1-minute sampling frequency, with 17478 observations and 7 features whereas the LSTM-RF was trained appliances and weather data with 1-hour sampling frequency that is equivalent to 3648 observations and 12 features.

We used the laptop computer with i7, @2.6 GHz, and Python processing algorithm to implement the load forecast and the results are shown in table 7.3 and figure 7.6. The combinatorial optimization (CO) training and testing took longer (29minutes) than the ensemble learning (LSTM+RF) based load forecast (13 minutes). The total computational time is equivalent to 42 minutes and this justifies that the proposed load forecasting framework can be implemented with a portable personal computer without the need for cloud computing that increases in privacy concerns of household consumption data.

Table 7.3: Computation time of load forecasting models based on load disaggregation.

time taken	CO	LSTM+RF	CO+LSTM+RF
Computation time(train+testing)	29 min	13 min	42 min

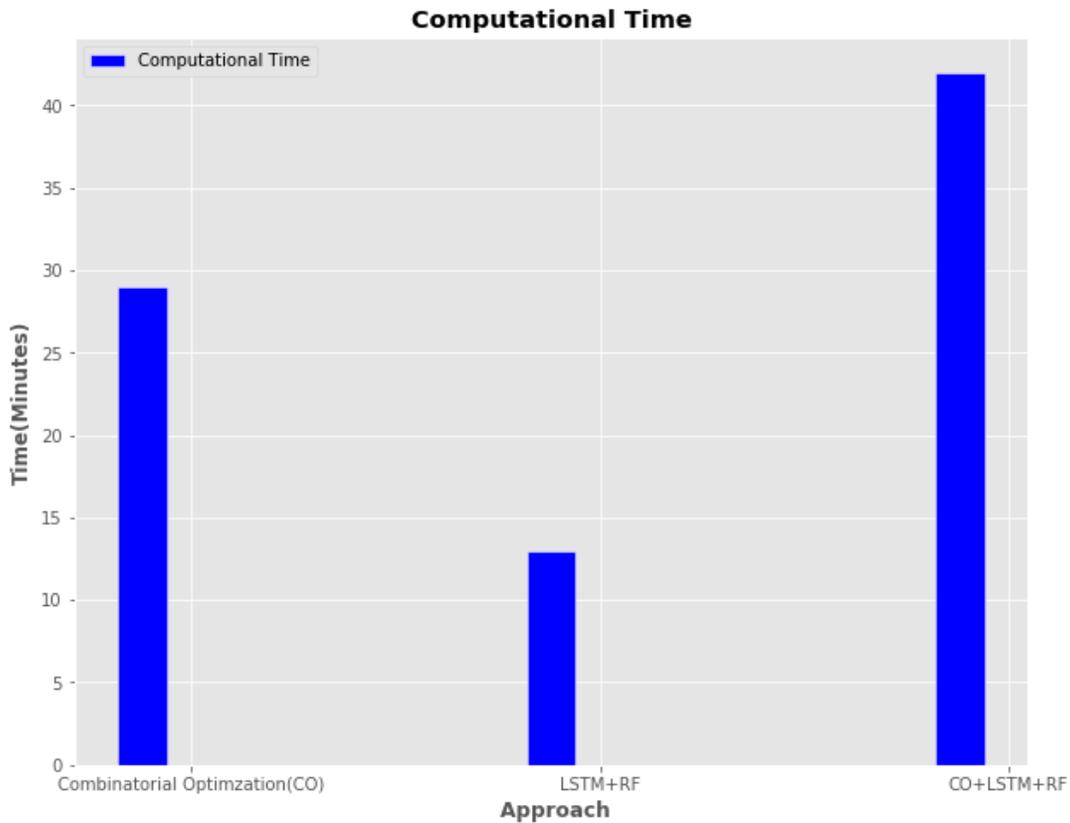


Figure 7.7:Computational time of load forecasting models based on load disaggregation

### 7.3 Complexity analysis

In addition to the weather data, this thesis added the appliances level data in the training set. Therefore, it is required to evaluate how the load forecast can handle the stress caused by the increased data (i.e. number of input features). We compared the performance of LSTM-RF ensemble learning trained with weather data (i.e. 5 features); appliances data (i.e. 7 features) and combined appliances data and weather data (i.e. 12 features). As shown in table 7.4 and figure 7.8 the model performance performed well in all 3 cases with the root mean square error in the range of 0.0624-0.0411.

As shown in 7.9 the model that includes disaggregated appliances data was able to reduce the forecasting error between the actual house total power and predicted house power demand for 3 days ahead as compared to the ensemble learning that was trained with weather data only. Moreover, the computational time for ensemble learning with weather and appliances data is almost twice that of a single model with weather and appliances data. The high forecasting error and computation time is obtained with the model trained with weather data and this could be originated to the variance among weather information.

Table 7.4: Computation time of load forecasting model based on number of features in the training stage.

	LSTM+RF: With weather data (5 features)	LSTM+RF: With Appliances data (7 features)	LSTM+RF: With weather and Appliances data (12 features)
NRMSE	0.0624	0.0421	0.0411
Computation time(train+testing)	6 min	5 min	13 min

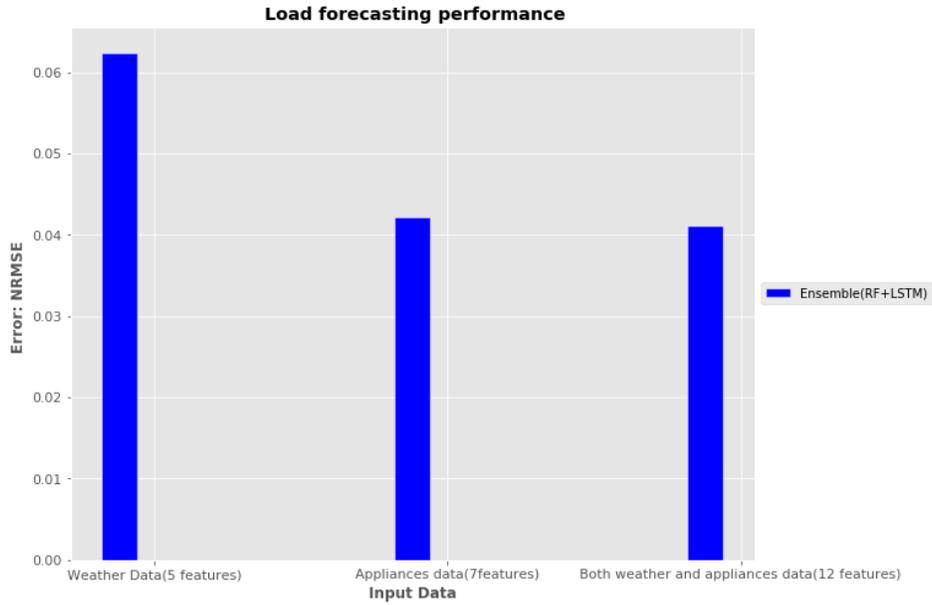


Figure 7.8: NRMSE error for load forecasting model based on number of input data (features).

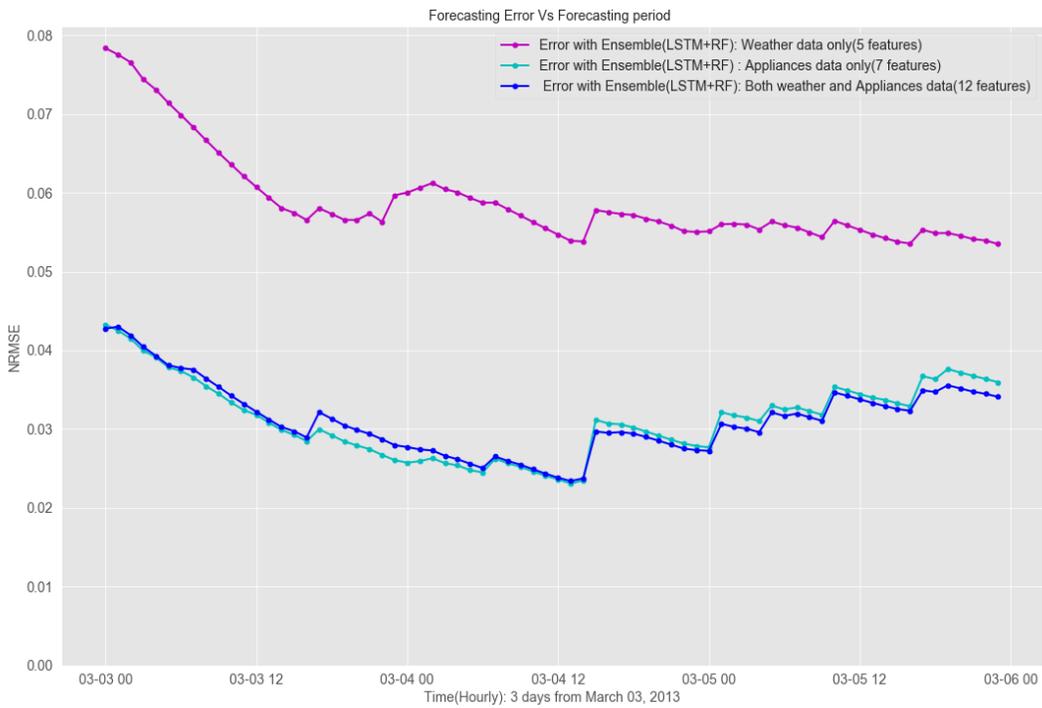


Figure 7.9: NRMSE error for load forecasting model based on forecasting length.

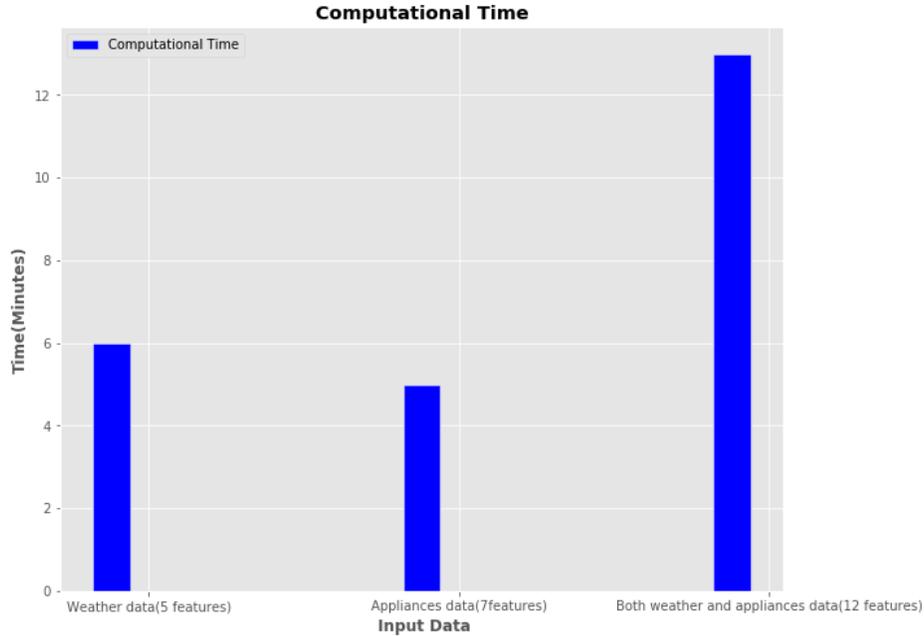


Figure 7.10: Computational time for load forecasting based on number of input data (features).

#### 7.4 Comparison of results

The performance of the combinatorial optimization (CO) based load disaggregation approach is compared with the approaches that were trained on the same dataset and reported in the literature [3] and [51]. Table 7.5 and figure 7.11 illustrate this comparison in terms of F1 score and it can be seen that the used CO has shown better disaggregation for two-state appliances such as heat pump but provided less disaggregate score for multi-state appliances such as the fridge. In terms of F1 score, the CO was able to disaggregate the heat pump with 0.92 score compare to 0.65 of denoising autoencoder [3] and 0.53 of long short term memory network[52]. The low CO disaggregation score for the fridge is originated from the fact that the CO model was trained to disaggregate signal features of appliances from aggregated energy consumption of a high number of appliances as opposed to a denoising autoencoder(DAE) and long short term memory(LSTM)

architecture. In the CO training, the aggregated house power was constituting with energy consumption of 8 appliances whereas DAE and LSTM used 6 and 5 appliances in the aggregated total, respectively.

In order to demonstrate the effectiveness of the proposed load forecast, the performance of the proposed ensemble learning(LSTM-RF) based load forecast was compared with the obtained results with other reference papers[6,12,53,54,55].

We first compared the performance of the proposed LSTM-RF with other forecasting methods [12,53] that were tested on the same AMPDs dataset. As shown in table 7.6 and figure 7.12.(a), the proposed LSTM-RF ensemble learning outperforms the two methods for 1 day ahead load forecast. In terms of mean percentage error (MAPE) metric, the LSTM-RF has achieved 31.6% compared to 34.46% of the feed-forward neural network [12] and 39.39% of autoregressive integrated moving average [53].

Secondly, in terms of data pre-processing based on combinatorial optimization. We compare the results of the proposed LSTM-RF model results with reference paper [6]. As shown in table 7.8and figure 7.12(b), the CO+LSTM+RF model was able to achieve low forecasting error compared to CO+FFNN [6]. For example, for 3 days ahead, the CO+LSTM+RF model achieved a room mean square error of 0.0386 compared to 0.0877 of [6].

Finally, to further evaluate the performance of the proposed CO+LSTM+RF model, we compared its performance with the performance of other works of literature [54,55] that were tested with other different appliances dataset. As it is presented in table 7.6 and figure 7.12. (b) the proposed method provides a lower RMSE as compared to the probabilistic

method presented in reference paper [54] and its results are much closer to that of the deep learning-based approach presented in reference [55].

Table 7.5: F1 score comparison on disaggregating appliances' energy consumption present in AMPDs dataset using different load disaggregation-based approaches.

(CO: Combinatorial optimization; DAE: denoising autoencoder; LSTM: Long short-term memory).

Performance (F1 score) comparison of load disaggregation-based approaches on AMPDs dataset.			
Appliance	This study: CO based pattern similarity.	Bonfiglie et al. [3]: DAE	Wang et al. [51]: LSTM
Heat pump	0.92	0.65	0.53
Fridge	0.43	0.39	0.38
Cloth dryer	0.52	0.91	0.98

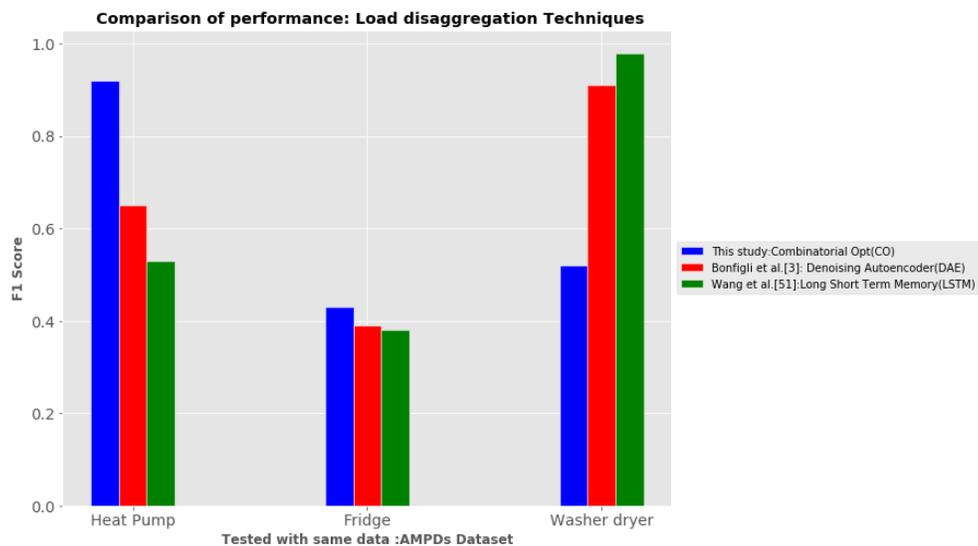


Figure 7.11: F1 score comparison on disaggregating appliances' energy consumption present in AMPDs dataset using different load disaggregation-based approaches.

Table 7.6: MAPE and RMSE comparison for different load forecasting approaches tested on different dataset.

(FF-NN: Feed forward neural network; FF-DNN: Feed forward deep neural network; CHSHMM: Conditional hidden Semi Markov Model; ARIMA: Autoregressive integrated moving average).

<b>Performance comparison of load forecasting-based load disaggregation</b>						
Study	Approach	Input data	Output	Forecast period	RMSE	MAPE (%)
This study	CO+LSTM+RF	Disaggregated appliance data (AMPds) and weather	House total power demand	3 days	0.0386	31.6
Ebrahim et al. [6]	CO+FF-NN	Disaggregated appliances data (UK-DALE [20])	House total power demand	1 day	0.0877	n/a
Din et al. [54]	FF-DNN	Actual appliances and weather	House total power demand	1 day	0.03	n/a
Ji et al. [53]	CHSHMM	Actual appliances data	Air conditioner Power demand	6 h	0.697	n/a
Kong et al. [11]	FF-NN	Actual appliances AMPs data	House total power demand	1 day	n/a	34.46
Gajowniczek et al. [52]	ARIMA	Actual appliances AMPs data	House total power demand	1 day	n/a	39.39

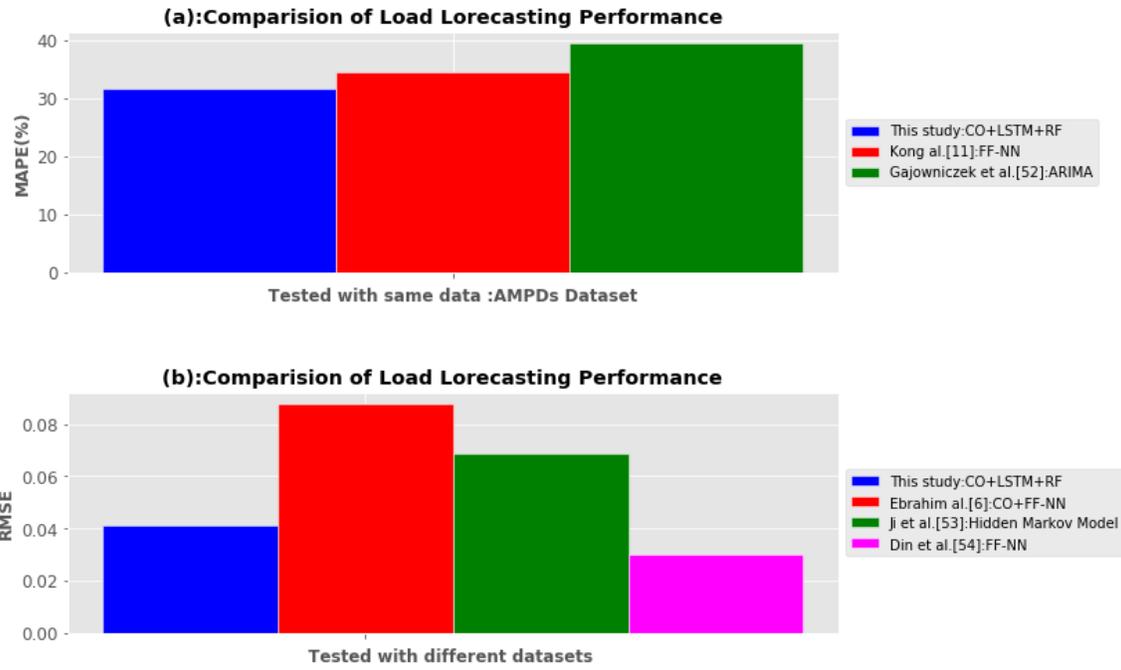


Figure 7.12: MAPE and RMSE comparison for different load forecasting approaches tested on different dataset

## CHAPTER 8: CONCLUSION AND FUTURE WORK

This covers the conclusion deduced from the thesis development and future work.

### 8.1. Conclusion

This thesis presented the short-term load forecast framework that utilizes the advantages of load disaggregation and ensemble learning for forecasting the power demand of a small house that is characterized by varying power demand due to human activities and weather conditions.

We experimented and compared the performance of the latest load disaggregation-based machine learning and deep learning techniques. The comparative results revealed that if trained with data from a small number of appliances, the optimization and probabilistic techniques can perform better for separating the ON/OFF type of appliances than recurrent and concurrent deep learning methods. For example, when trained with 8 appliances data, in terms of F1 score, the Combinatorial optimization was able to disaggregate the heat pump with 0.92 scores compare to 0.65 of denoising autoencoder and 0.53 of long short-term memory network. Interestingly, the deep learning-based methods succeed with disaggregating the appliances with complex signature features. For example, the denoising autoencoder network achieved over 89% score for the disaggregating fridge and 80% for washer dryer.

The forecasting model that combined the benefits of load disaggregation, weather data, and ensemble learning has been designed to improve the short-term load forecast. The developed load forecasting is based on the staking principle, in which the hybrid ensemble learning utilized the potential of long short-term memory and random forest (LSTM-RF)

to obtain more robust predictions and predicts the house power demand for three days ahead.

The simulation results indicated that the performance of individual predictor was greatly improved by including the estimated appliances energy consumption in the model training. For three days ahead load forecasts, the model with estimated appliances level data was able to sustain the low ( $> 0.05$ ) normalized root mean square error (NRMSE) as opposed to the model with weather data only that achieved the error in the 0.05-0.08 range. Although each predictor achieved the reduced forecasting error, the LSTM-RF ensemble outperformed the single models for three days ahead period. In terms of NRMSE, the LSTM-RF ensemble learning reduced the error up to 0.0411 compared to 0.0706 of long short-term memory (LSTM) and 0.0423 of Random forecast (RF).

The computation time of the ensemble learning is found to be double of a single model and is within the acceptable limit for providing energy feedback using a local computer system. This could help in eliminating the need to use cloud computing and reduce the privacy concerns on client energy usage data.

The performance of the proposed LSTM-RF ensemble learning is compared with other approaches that were trained on the same dataset. The results have shown that the LSTM-RF based on appliance features extraction and weather data can significantly improve the forecasting compared to a single method for one day ahead load forecasting. In terms of mean percentage error (MAPE) metric, the LSTM-RF has achieved 31.6% compared to 34.46% of feed-forward neural networks and 39.39% of the autoregressive integrated moving average.

The volatility in daily house energy consumption implicates that the energy clients need the accurate short-term load forecast. This study proved that home energy consumption is related to residents' activities and more energy is consumed in afternoons as all family members are usually at home to use most of the appliances. For load disaggregation practice, the results show that the combinatorial optimization could disaggregate the signal features of ON/OFF type of appliances with over 80 scores.

For load forecasting based on load disaggregation and weather information, the results showed that the appliances' signal features are more important to the house energy predictions than meteorological weather conditions. Moreover, the hybrid methods can contribute to reducing the forecasting error compared to the single forecasting method.

## **8.2. Future Work**

The thesis ends with two limitations that could be addressed to further provide useful feedbacks for home residents and the short-term load forecast. Firstly, the combinatorial based load disaggregation was trained and tested with data from a single house in Canada, which limits the model applicability for separating appliances in other houses in Canada and scalability for separating other houses in other countries. The future work could consider testing the disaggregation model with other datasets with appliance level data of at least two houses or datasets from other countries. This could help a lot because appliances' specifications may differ from one to another.

Secondly and finally, the load forecast architecture has integrated the long short-term memory and random forest algorithms without utilizing ensemble pruning, optimizing, and tuning hyperparameters. In future work, the ensemble pruning can be achieved by combining

different evolutionary algorithms with random learners. For example, the particle swarm optimization algorithm can be combined with support vector regression to obtain the optimized parameters of support vector regression. To further reduce the error of the short-term load forecasting, different feature engineering techniques can be utilized to obtain and add categorical features such hours of the day and days of the week in the training dataset.

## Bibliography

- [1].H. T. Nguyen, D. T. Nguyen and L. B. Le, "Energy Management for Households with Solar Assisted Thermal Load Considering Renewable Energy and Price Uncertainty," in *IEEE Transactions on Smart Grid*, vol. 6, no. 1, pp. 301-314, Jan. 2015.
- [2].J. Kim, T.-T.-H. Le, and H. Kim, "Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature," *Computational Intelligence and Neuroscience*, vol. 2017, pp. 1–22, 2017.
- [3].R. Bonfigli, S. Squartini. *Machine Learning Approaches to Non-Intrusive Load Monitoring*. Springer Briefs in Energy.2020
- [4].J. Kelly and W. Knottenbelt, "Neural NILM," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments - BuildSys '15*, New York, New York, USA, 2015, pp. 55–64, ACM Press.
- [5].V. Dehalwar, A. Kalam, M. L. Kolhe and A. Zayegh, "Electricity load forecasting for Urban area using weather forecast information," *2016 IEEE International Conference on Power and Renewable Energy (ICPRE)*, Shanghai, 2016, pp. 355-359
- [6].A. F. Ebrahim and O. A. Mohammed, "Pre-processing of energy demand disaggregation-based data mining techniques for household load demand forecasting," *Inventions*, vol. 3, no. 3, pp. 2411–5134, 2018.
- [7].Fischer, "Feedback on household electricity consumption: a tool for saving energy?" *Energy efficiency*, vol. 1, no. 1, pp. 79–104, 2008.

- [8]. S. Darby, "The effectiveness of feedback on energy consumption", A Rev. DEFRA Literature Metering Billing Direct Displays, vol. 486, no. 2006, pp. 26, 2006
- [9]. C. Armel, "Energy disaggregation, precourt energy efficiency center," Stanford University Lecture, 2011.
- [10]. M. D'Incecco, S. Squartini and M. Zhong, "Transfer Learning for Non-Intrusive Load Monitoring," in *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1419-1429, March 2020
- [11]. W. Kong, Z. Y. Dong, D. J. Hill, F. Luo and Y. Xu, "Short-Term Residential Load Forecasting Based on Resident Behaviour Learning," in *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 1087-1088, Jan. 2018.
- [12]. P. P. M. d. Nascimento, Applications of Deep Learning Techniques on NILM, Ph.D. thesis, COPPE/UFRJ, 2016.
- [13]. H. Rashid, P. Singh and A. Singh, "I-BLEND a campus-scale commercial and residential buildings electrical energy dataset", *Sci. Data*, vol. 6, no. 1, Mar. 2019.
- [14]. N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, et al., "Nilmtk: An open source toolkit for non-intrusive load monitoring", *Proc. 5th Int. Conf. Future Energy Syst.*, pp. 265-276, 2014
- [15]. L. M. Candanedo, V. Feldheim and D. Deramaix, "Data driven prediction models of energy use of appliances in a low-energy house", *Energy Buildings*, vol. 140, pp. 81-97, Apr. 2017.

- [16]. J. Z. Kolter and M. J. Johnson, "REDD: A Public Data Set for Energy Disaggregation Research," in SustKDD workshop on Data Mining Applications in Sustainability, 2011, pp. 1–6.
- [17]. D. Murray, L. Stankovic and V. Stankovic, "An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study", *Sci. Data*, vol. 4, no. 1, Jan. 2017.
- [18]. S. Makonin, F. Popowich, L. Bartram, B. Gill and I. V. Bajic, "AMPds: A public dataset for load disaggregation and eco-feedback research", *Proc. IEEE Electr. Power Energy Conf.*, pp. 1-6, Aug. 2013.
- [19]. S. Makonin, "HUE: The hourly usage of energy dataset for buildings in British Columbia", *Data Brief*, vol. 23, pp. 103744, Apr. 2019.
- [20]. J. Kelly and W. Knottenbelt, "The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes," *Scientific Data*, vol. 2, 3 2015.
- [21]. S. Makonin, B. Ellert, I. V. Bajić and F. Popowich, "Electricity water and natural gas consumption of a residential house in Canada from 2012 to 2014", *Sci. Data*, vol. 3, no. 1, pp. 1-12, Jun. 2016.
- [22]. R. Platon, V. R. Dehkordi and J. Martel, "Hourly prediction of a building's electricity consumption using case-based reasoning artificial neural networks and principal component analysis", *Energy Buildings*, vol. 92, pp. 10-18, Apr. 2015

- [23]. ASHRAE, ASHRAE Guideline 14: Measurement of Energy and Demand Savings, ASHRAE, Atlanta, 2002.
- [24]. L. Qiuyu et al., "Short-term load forecasting based on load decomposition and numerical weather forecast," 2017 IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, 2017, pp. 1-5
- [25]. Y. Ma, Q. Zhang, J. Ding, Q. Wang and J. Ma, "Short Term Load Forecasting Based on iForest-LSTM," *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Xi'an, China, 2019, pp. 2278-2282
- [26]. M. A. Khairalla, X. Ning, N. T. Al-Jallad and M. O. El-Faroug, "Short-term forecasting for energy consumption through stacking heterogeneous ensemble learning model", *Energies*, vol. 11, no. 6, pp. 1605, 2018
- [27]. O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Using hidden markov models for iterative non-intrusive appliance monitoring," in *Neural Information Processing Systems, Workshop on Machine Learning for Sustainability*, Dec. 2011.
- [28]. A. Harell, S. Makonin and I. Bajic, "WaveNILM: A causal neural network for power disaggregation from the complex power signal", *Proc. 44th Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, pp. 8335-8339, 2019.
- [29]. H. Rafiq, H. Zhang, H. Li and M. K. Ochani, "Regularized LSTM Based Deep Learning Model: First Step towards Real-Time Non-Intrusive Load Monitoring," *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, Oshawa, ON, 2018, pp. 234-239

- [30]. M. Z. A. Bhotto, S. Makonin, and I. V. Bajić, "Load disaggregation based on aided linear integer programming," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 7, pp. 792–796, July 2017.
- [31]. P. B. M. Martins, J. G. R. C. Gomes, V. B. Nascimento and A. R. de Freitas, "Application of a deep learning generative model to load disaggregation for industrial machinery power consumption monitoring", 2018 IEEE International Conference on Communications Control and Computing Technologies for Smart Grids (SmartGridComm), pp. 1-6, Oct 2018.
- [32]. A. Rodriguez-Silva and S. Makonin, "Universal non-intrusive load monitoring (UNILM) using filter pipelines probabilistic knapsack and labelled partition maps", *Proc. IEEE PES Asia–Pacific Power Energy Eng. Conf. (APPEEC)*, pp. 1-6, Dec. 2019.
- [33]. L. Mauch and B. Yang, "A new approach for supervised power disaggregation by using a deep recurrent LSTM network", *IEEE Glob. Conf. Signal Inf. Process*, pp. 63-67, 2015.
- [34]. D. Murray, L. Stankovic, V. Stankovic, S. Lulic and S. Sladojevic, "Transferability of neural network approaches for low-rate energy disaggregation", *Proc. Int. Conf. Acoust. Speech Signal Process.*, pp. 8330-8334, 2018.
- [35]. L. Mauch and B. Yang, "A novel DNN-HMM-based approach for extracting single loads from aggregate power signals", *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 2384-2388, 2016.

- [36]. B. Humala, A. S. U. Nambi, and V. R. Prasad, "Universal-NILM: A Semi-supervised Energy Disaggregation Framework using General Appliance Models," in Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18, New York, New York, USA, 2018, pp. 223–229, ACM Press.
- [37]. J. Z. Kolter and T. Jaakkola, "Approximate inference in additive factorial HMMs with application to energy disaggregation", J. Mach. Learn. Res. Proc. Track, vol. 22, pp. 1472-1482, Apr. 2012.
- [38]. S. Makonin, F. Popowich, I. V. Bajić, B. Gill and L. Bartram, "Exploiting HMM sparsity to perform online real-time nonintrusive load monitoring", IEEE Transactions on Smart Grid, vol. 7, no. 6, pp. 2575-2585, 2016.
- [39]. M. J. Johnson and A. S. Willsky, "Bayesian nonparametric hidden semi-Markov models", J. Mach. Learn. Res., vol. 14, no. 1, pp. 673-701, 2013.
- [40]. N. Batra, H. Dutta and A. Singh, "INDiC: Improved non-intrusive load monitoring using load division and calibration", Int. Conf. Mach. Learn. Appl, pp. 79-84, 2013.
- [41]. L. Breiman, "Random Forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
- [42]. R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee, "Boosting the margin: A new explanation for the effectiveness of voting methods," *Ann. Stat.*, vol. 26, no. 5, pp. 1651–1686, 1998
- [43]. S.V.Maiya, Data Science incentive Capstone Project(2018), GitHub repository, <https://github.com/shashankvmaiya/Bike-Sharing-Demand-Prediction>

- [44]. A.Ahmed, Introduction to data science(2018), GitHub repository, <https://github.com/arslan2k12/Household-Power-Consumption-Analysis-and-Forecasting>
- [45]. L. Pereira and N. Nunes, "Performance evaluation in non-intrusive load monitoring: Datasets metrics and tools-A review", Wiley Interdiscip. Rev. Data Min. Knowl. Discov., pp. 1-17, 2018.
- [46]. L. Wang, X. Luo and W. Zhang, "Unsupervised energy disaggregation with factorial hidden Markov models based on generalized backfitting algorithm", Proc. IEEE Region Conf., pp. 1-4, Oct. 2013.
- [47]. 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 Trans. Smart Grid, pp. 1-8, Sep. 2017.
- [48]. M. Gaur and A. Majumdar, "Disaggregating Transform Learning for Non-Intrusive Load Monitoring," in *IEEE Access*, vol. 6, pp. 46256-46265, 2018.
- [49]. S. Kumar, L. Hussain, S. Banajee and M. Reza, "Energy load forecasting using deep learning approach-LSTM and GRU in spark cluster", Proc. EAIT, pp. 1-4, Jan. 2018.
- [50]. J. Chavat, J. Graneri, and S. Nesmachnow, "Household energy disaggregation based on pattern consumption similarities," in Ibero- American Congress on Information Management and Big Data (ICSCCITIES 2019), pp. 54–69, 2019.

- [51]. T. S. Wang, T. Y. Ji and M. S. Li, "A New Approach for Supervised Power Disaggregation by Using a Denoising Autoencoder and Recurrent LSTM Network," 2019 IEEE 12th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED), Toulouse, France, 2019, pp. 507-512.
- [52]. K. Gajowniczek and T. Zabkowski, "Electricity forecasting on the individual household level enhanced based on activity patterns", PloS one, vol. 12, no. 4, pp. 174-198, 2017.
- [53]. Y. Ji, E. Buechler and R. Rajagopal, "Data-Driven Load Modeling and Forecasting of Residential Appliances," in *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2652-2661, May 2020
- [54]. G. Mohi Ud Din, A. U. Mauthe and A. K. Marnerides, "Appliance-level short-term load forecasting using deep neural networks", Proc. Int. Conf. Comput. Netw. Commun. (ICNC), pp. 53-57, Mar. 2018.