Federated Learning Framework and Energy Disaggregation Techniques for Residential Energy Management

Shamisa Kaspour

December 2022

Abstract

Residential energy use is a significant part of total power usage in developed countries. To reduce overall energy use and save funds, these countries need solutions that help them keep track of how different appliances are used at residences. Non-Intrusive Load Monitoring (NILM) or energy disaggregation is a method for calculating individual appliance power consumption from a single meter tracking the aggregated power of several appliances. To implement any NILM approach in the real world, it is necessary to collect massive amounts of data from individual residences and transfer them to centralized servers, where they will undergo extensive analysis. The centralized fashion of this procedure makes it time-consuming and costly since transferring the data from thousands of residences to the central server takes a lot of time and storage. This thesis proposes utilizing Federated Learning (FL) framework for NILM in order to make the entire system cost-effective and efficient. Rather than collecting data from all clients (residences) and sending it back to the central server, local models are generated on each client's end and trained on local data in FL. This allows FL to respond more quickly to changes in the environment and handle data locally in a single household, increasing the system's speed. On top of that, without any data transfer, FL prevents data leakage and preserves the clients' privacy, leading to a safe and trustworthy system. For the first time, in this work, the performance of deploying FL in NILM was investigated with two different energy disaggregation models: Short Sequence-to-Point (Seq2Point) and Variational Auto-Encoder (VAE). Short Seq2Point with fewer samples as input window for each appliance, tries to simulate the real-time energy disaggregation for the different appliances. Despite having a light-weighted model, Short Seq2Point lacks generalizability and might confront some challenges while disaggregating multi-state appliances. VAE is a generative model with a complex structure, which resolves the mentioned issues in Short Seq2Point. The proposed experiments are examined using two real-life datasets of appliance-level power from the UK: UK-DALE and REFIT. In order to more thoroughly examine the utility of FL in this study, several experiments have been

conducted, including the implementation of attention-based aggregation in FL, and the addition of differential privacy noise to the aggregated parameters. The attention-based mechanism in FL, is a novel aggregation approach that applies a scale factor to the parameters of each client based on the importance of the information it carries. The results were then compared with recent cutting-edge studies in the same field. Based on the results, this study presents that FL framework provides comparable performance to its centralized counterparts while enhancing clients' privacy.

Acknowledgements

I would like to express my deepest appreciation to my supervisor, Dr. Abdulsalam Yassine, for his guidance and patience in helping me select my intended field of study and stay focused on what I enjoy learning. His continued attention and assistance with my research and studies over the last two years have inspired me to learn more and be more creative. In addition, the financial aid he provided for me as a research assistant enabled me to be less distracted by the costs of living and tuition during my master's program.

Furthermore, I am grateful for my family's unwavering love and support, especially my spouse, who has always encouraged me to stay eager to learn new things. Despite the fact that moving to a new country to pursue my education was a significant challenge for me, my husband's emotional support kept me optimistic about the future and allowed me to see every problem as an opportunity.

Lastly, special thanks to Lakehead University and the department of computer science for various entrance awards as well as the graduate assistantship that enabled me to pay for my educational expenses in Canada. Furthermore, I am deeply indebted to the Vector Institute committee for selecting me as a qualifying student for the Artificial Intelligence scholarship at Lakehead University, which eased my journey during this two-year course of study.

Contents

List of Figures

List of Tables

Acronyms

AMI Advanced Metering Infrastructures. 7, 10

ARIMA Autoregressive Integrated Moving Average. 7, 35, 39

ATL Appliance Transfer Learning. 7, 41

BEMS Building Energy Management System. 7, 35

CART Classification and Regression Tree. 7, 39

CER Commission for Energy Regulation. 7, 35

CNN Convolutional Neural Network. 7, 20, 29, 36, 38, 41, 42, 47, 55

CTL Cross-domain Transfer Learning. 7, 41

DP Differential Privacy. 1, 3–5, 7, 15, 31, 32, 37, 40, 58, 62, 71, 72, 74, 76

DP-FL Differential Privacy based Federated Learning. 7, 15, 32, 37

DSM Demand Side Management. 7, 34

Fed-SMP Federated Sparsified Model Perturbation. 7, 32, 37, 76

FedAtt Federated Attention-based. 1, 4, 6, 7, 12, 14–18, 26, 27, 40, 55–58, 61–63, 66–76

FedAvg Federated Averaging. 1, 4, 6, 7, 11, 12, 15–17, 24, 28, 29, 31, 36, 37, 40, 42, 48–51, 53–55, 58, 61–63, 66, 68–71, 73–76

FL Federated Learning. 1–5, 7, 11–18, 24–32, 36, 37, 42, 43, 46, 49–51, 54, 55, 58, 66–68, 73–76

GRU Gated Recurrent Units. 7, 11, 14, 31, 36, 38, 42, 65

IID Independent and Identically Distributed. 7, 29

KL Kullback-Leibler. 7, 22

LSTM Long-Short Term Memory. 7, 11, 14, 29, 36, 38, 41

MAE Mean Absolute Error. 7, 22, 50, 68

MLP Multi-Layer Perceptron. 7, 29, 33, 36

NILM Non-Intrusive Load Monitoring. 1, 2, 4, 5, 7, 10–20, 24–26, 28, 30–32, 38, 40–43, 45, 49–51, 54, 55, 58, 65, 66, 73–76

NLP Natural Language Processing. 7, 12, 55, 65

OLIVE Oblivious and Differentially Private Federated Learning on Trusted Execution Environment. 7, 32, 37, 76

REDD Reference Energy Disaggregation Dataset. 7, 28, 33, 36, 38

ReLU Rectified Linear Unit. 7, 66

RETE Relative Error in Total Energy. 7, 22, 61

RMSP Root Mean Square Propagation. 7, 67

RNN Recurrent Neural Network. 7, 11, 38, 41, 55

Seq2Point Sequence-to-Point. 1, 2, 4–7, 10–12, 14, 16–18, 20, 21, 28, 36, 40–43, 46–51, 53, 55, 56, 61–63, 65, 66, 68, 70, 73, 74

Seq2Seq Sequence-to-Sequence. 7, 20, 38, 41, 55, 65

SVM Support Vector Machines. 7, 33

TL Transfer Learning. 3, 7, 28, 36, 41, 74, 75

UK-DALE UK Domestic Appliance Level Electricity. 1, 2, 4–7, 17, 28, 33, 34, 36, 38, 40, 43–45, 48, 50, 51, 55, 56, 61, 62, 64, 69–74

UKERC-EDC UK Energy Research Centre Energy Data Centre. 7, 43

VAE Variational Auto-Encoder. 1, 2, 4, 6, 7, 10–12, 15, 17, 18, 21, 40, 65–72, 74

Chapter 1

Introduction

1.1 Introduction

Reduction in energy waste necessitates understanding how energy is consumed, leading to better energy management and, ultimately, more efficient energy usage [53]. As a result, energy management has become a growing study subject in recent years, attracting particular attention from the machine learning community. Households are one of the most significant sources of energy waste [54]. To monitor the energy consumption of a house, it is necessary to deploy some smart sensors for each appliance to extract the pattern of its energy usage and record its power consumption [54, 57]. However, installing all these sensors on appliances is intrusive, expensive, and hard to maintain [60]. Therefore, leading us to use another method for energy monitoring, such Non-Intrusive Load Monitoring (NILM) [21]. Energy disaggregation, also known as NILM, monitors appliance-level power consumption by breaking down the overall electric load. NILM evaluates the overall electric load of houses by installing some Advanced Metering Infrastructures (AMI) at the electrical entry to measure signals such as active and reactive power, voltage, current, etc. For NILM to be applicable in a real-world scenario, the aggregate power of each household and its appliances should be visible in this technique.

Studies on NILM, shows that deep neural networks most often perform well regarding disaggregating the consumed power of each appliance [32, 68]. Two recent proposed methods in this area are Short Sequence-to-Point (Seq2Point) [32] and Variational Auto-Encoder (VAE) [33]. The Short Seq2Point method is a sliding window real-time energy disaggregation technique that uses a window of aggregate energy as input to forecast the midpoint value of appliance consumption within the same window. The original Seq2Point [68] takes one hour of data as the input, making it an unsuitable model for online

energy disaggregation. On the other hand, Short Seq2Point reduces the input window size to 5–20 minutes, which translates into a shorter latency between the results and their presentation. The results of [32] demonstrates that compare to other NILM methods such as Long-Short Term Memory (LSTM), Recurrent Neural Network (RNN) and Gated Recurrent Units (GRU), Short Seq2Point mainly performs better.

Despite providing a lightweight model with the ability for online energy disaggregation, some work remains to be done regarding multi-state appliances and the ability of generalization to various residences in Short Seq2Point. The authors of [33] discuss these problems and suggest a VAE strategy for energy disaggregation. With the help of the probabilistic encoder, this technique provides a powerful model for encoding the aggregate data to reconstruct the desired appliance's consumption. For multi-state appliances, the suggested approach accurately creates more detailed load profiles, leading to better power signal reconstruction. The model's enhanced ability to generalize across distinct residences is partially due to its regularized latent space. The results of this paper [33] show that this method can increasingly improve the performance of load monitoring.

While the aforementioned energy disaggregation strategies have made significant progress toward separating the power consumed by a single appliance from the aggregate power, it is still inefficient and time-consuming to move data from each residence to the central server to run NILM methods on it. Additionally, the storage cost required to maintain this data on the central server is expensive. To tackle these problems, for the first time in this thesis, a distributed machine learning framework called Federated Learning (FL) has been deployed for NILM methods mentioned earlier: Seq2Point and VAE.

FL is a distributed machine learning framework that trains a model across numerous decentralized edge devices called clients, keeping local data without transferring it. It was first introduced for auto-correction in mobile keyboards by google [20, 50]. In this framework, only the parameters of each client's local model will be sent to the central server. The central server has a global model with the same structure as the local ones. There are no training epochs in the global model, and the parameters of it will be updated with an aggregated form of all the local parameters. The global model will then be evaluated with data that has not been used in any training epochs of local models. In our work, the performance of FL with two different NILM approaches will be investigated in comparison with their centralized counterparts to prove the feasibility of using this framework in real-world applications.

There are several aggregation methods in FL framework. However, the only one that was experimented in NILM is Federated Averaging (FedAvg) [12, 35, 63, 25]. FedAvg is a way of aggregating

the local parameters by calculating their average value. First, in each client, parameters within local models will be multiplied by a scale factor indicating the proportion of local data over all training data used in local models. Then, an average of all the scaled parameters of the local models will be used for the global model. This aggregating method along with Short Seq2Point model was experimented in our paper [25].

The simplicity of FedAvg makes it easy to use, yet, it may not provide accurate results. Clients' energy consumption can be dramatically impacted by their individual traits. The amount of time people spend using various appliances is directly related to their preferences regarding those devices. Furthermore, the number of people living in a household, the location of the house, the type of dwelling, and a variety of other factors can all have an impact on the total amount of energy consumed by a household. In this work, for the first time, a novel aggregation approach called Federated Attention-based (FedAtt) is used for NILM. FedAtt, which was proposed for Natural Language Processing (NLP) in [23], tries to make global parameters as similar as possible to the local ones. During this technique, each parameter in the local model will gain a scale factor that indicates the local data's importance. By using FedAtt, FL can provide a more generalized framework that considers the attributes of each client. In this study, the FedAtt is applied on both Seq2Point and VAE models to compare their performance with FedAvg.

Besides reducing the costs and time in FL framework, it provides an additional privacy to the system. Still, a privacy-preserving approach, such as differential privacy, can be utilized to protect the local parameters from the inverse attack. In this method, a scaled randomized noise can be added to the local parameters during the federated optimization step. In our work, differential privacy noise is also added in both FedAvg and FedAtt to illustrate how FL system can perform.

1.2 Problem Definition and Motivation

In light of the rapidly expanding global population and the increasing number of residential buildings, energy consumption is steadily increasing. A significant increase in the amount of wasted energy accompanies the escalating level of energy consumption. Therefore, in developed countries, it is of the utmost importance to monitor the appliances that are used in households to provide a useful pattern for customers and help cut down on energy waste.

NILM, is a method that helps identify the active appliances in a household based on the total power consumption of all the appliances. For an energy disaggregation system to work effectively, it must monitor many residences to understand how energy consumption can be managed. Taking into account

the number of appliances that each person has in their own house, there would be an impossibly vast quantity of data produced by these devices, in addition to a significant amount of electricity that would be consumed by them. The ever-increasing amount of data presents the most significant obstacle that must be overcome. It is challenging to perform tasks in real-time since exchanging the aggregate power and all the other signals related to multiple appliances from numerous households to central servers takes a long time. Although, there are many big data approaches to ease this process, preventing from data exchange is not a case in any of them. In addition, managing all of this data in the central servers is a sophisticated process that requires a significant amount of storage space to be available.

On top of that, personal data leakage and fraud are two major concerns people worldwide are dealing with nowadays. As a result of these two factors, the clients find the systems to be untrusted and unreliable. Clients are vested in ensuring that their data is secure and has not been moved from their devices to any other location.

All of the issues described in the previous paragraphs serve as the motivation for carrying out this study. The deployment of the FL framework and the recently suggested NILM models, as well as the examination of their performance, provide societies with the assurance that they can use this framework for real-world scenarios incorporating energy disaggregation. Changing from a centralized model to a distributed one in FL not only saves a significant amount of money and time in residential energy management but also produces a system that protects people's privacy. This indicates that FL system will be able to manage more clients, anticipate the high-consumption appliances, and create guidelines for the communities on how to operate their appliances to consume less electricity. The potential usages of FL and NILM in the industry can be predicting behavioral patterns of different clients in the system, load management in the pick of energy consumption, and energy supply and demand.

For aggregating the data in real life for FL and NILM, the aggregate power signal of each household should be monitored with a sensor with a specific sample rate. Preprocessing of the aggregate data will be done in the local models. To have test data for evaluating the system, it is necessary to monitor the appliances and aggregate power of a sample household in the same area. The local models should periodically inform the system about available appliances in the house. This way, the system performs better at disaggregating the data.

1.3 Technical Challenges

To move forward with the implementation of this research, different problems had to be solved. These challenges are as follows:

- Challenge 1 - Number of clients in FL: In this research, each household in a dataset represents a client in FL framework. The higher the number of clients, the more precise the resulting information will be. However, there is always a restricted number of clients in the available datasets. In our thesis, the proposed models are examined with a dataset that includes more houses for each appliance to better represent how FL performs with different ratios of clients in real-life energy disaggregation approaches. Furthermore, based on the number of clients and local models' epochs in FL, each global epoch might take a lot of time to finish. Such a framework has a complicated structure, resulting in many hyperparameters that can affect its performance. For these challenges, no one has done proper hyper-parameter tuning in any of the most recent publications in this field. This study attempts to identify the most critical parameters that affect FL performance and effectively tune the parameters for the newly proposed aggregation method fedatt.

- Challenge 2 - Inverse attack on local parameters in the central server: The parameters of the local models are the only data exchanged in the FL framework. Although adopting FL improves system security, there is still uncertainty about potential attacks on local parameters within the central server. These attacks might put clients' personal information at risk of leakage. Some methods are used to get around this challenge, like adding a differential privacy noise to the local parameters in the federated optimization step.

- Challenge 3 - Appliance energy use variations in different households: How people interact with their appliances might vary depending on various circumstances, including personal preferences, house structure, location, number of rooms, etc. As a result, the power signals produced by the same appliance in households may differ greatly. All the recent FL and NILM articles ignore this challenge and employ a fairly straightforward aggregation technique that averages all local parameters. To address this issue, we first employ an attention-based FL called FedAtt for the first time. This enables us to account for the diversity of our clients and produce a standardized system that functions for everyone.

- Challenge 4 - Lack of generalizability in disaggregation models: GRU, LSTM, and Seq2Point disaggregation models mostly struggle with disaggregating energy for multi-state appliances. This problem might also be apparent in the FL framework. As a solution, a relatively new disaggregation model called VAE is used in our framework. VAE is a generative model with a powerful

probabilistic encoder and several skip connections, which helps reconstruct the appliance's energy consumption from the input aggregate power.

1.4 Research Approach

This thesis aims to investigate the performance of deploying a distributed machine learning technique called FL with energy disaggregation models. Each step of the proposed approach for this research has been explained in the following paragraphs:

- In a real-world application of FL, numerous clients are involved simultaneously; thus, in our thesis, it is necessary to investigate the importance of the number of clients in the federated NILM, which is challenge 1 in this thesis. By using a dataset containing 20 houses and randomly choosing different fractions of clients from that for FL framework, this research tries to demonstrate the stability of FL performance regarding the number of clients.

- Although using a distributed machine framework such as FL, improves the clients' privacy in the system; still, the parameters of each client in the central server can be under inverse attack. During the training of a local model, the parameters of the model store information about the training data. This information from local parameters on the central server can risk the client's privacy. By adding a Differential Privacy noise to each vector of local parameters in the federated optimization step of FL, challenge 2 can be solved. This client-sided Differential Privacy based Federated Learning (DP-FL) technique was applied on both FedAvg and FedAtt.

- According to [56, 55], NILM does not provide clients with sufficient information regarding their electricity usage patterns. These patterns differ for each person, and they are essential to be noticed for energy disaggregation. They even can be different for the same appliance in different households. To tackle challenge 3, an Attention-based FL, also known as FedAtt, is used in this research. Since each client can have a unique energy usage pattern, the generated signal for his household's aggregated power will differ. With FedAtt, the central server tries to understand these differences in local models and update the global parameters in a way that effectively represents each client's characteristics. FedAtt is a new aggregation method for FL. Combining FedAtt with NILM has not been addressed in other studies before this one.

- Challenge 4, which is the lack of generalizability in the disaggregation models, was solved using a newly proposed NILM approach called Variational Auto-Encoder (VAE) [33]. VAE, consists of

an encoder and decoder, which generate complex load profiles to help the model understand the various features of the input signal. Also, its U-Net shape and skip connections help the decoder better reconstruct the desired appliance power signal with the attributes it learns from through these connections. This novel disaggregation model was utilized in two different aggregation approaches (FedAtt and FedAvg), proving that a generalized model can significantly alter the performance of the FL framework [33].

1.5 Contributions

The contributions of this thesis are:

- This research proposes a system that combines NILM and FL to collect data from appliances and analyze energy consumption. In the proposed system, modeling parameters are the only information transferred from the local models to the global one, which protects the privacy of information because household data remains in the local database. Thus, there is no need to take a backup from the massive data in the central server. This thesis suggested the use of the short version of the Seq2Point model instead of the original Seq2Point model, which makes it faster and less computationally intensive for utilizing the model in a real-world application. This thesis evaluated the model through rigorous experimentation and compared the proposed privacy-preserving distributed model results with the work in [32]. The results show that it is possible to achieve performance and accuracy in the same range as the centralized version without sacrificing the clients' privacy, which is a big step in the area of NILM. Short Seq2Point was examined with another dataset [41] with around 20 households to show the importance of the number of clients in FL for solving the challenge 1.

- For the first time, this research uses a novel aggregation method in FL (challenge 2), called FedAtt [23], presenting a way to consider the weights of a client that can have more impact on the global model's performance. Instead of averaging the weight of all the clients, now weights with more valuable information have higher priorities than the others. To determine the framework's optimal parameters, hyperparameter tuning was done for FedAtt (challenge 4).

- FL, as a decentralized machine learning method potentially, can improve the privacy and security of the clients whose data are used in the system. However, during the federated optimization step in the global model, the parameters of local models are still at risk of leakage. By adding different values of differential privacy noise to local model weights, this research indicates that in some

cases, FL still provides promising results comparable to the original centralized model outcomes (challenge 3). A shorter window size for data samples might make the NILM procedure faster with less computational intensity. Also, it makes it possible for the system to produce semi-realtime results. However, since appliances are mainly in the OFF state, shorter window size causes many instances that do not have the consumed power value in the ON state. Moreover, Short Seq2Point can not perform well on multi-state appliances like microwave. This thesis examines the larger window size of data samples with a new generalized model called VAE (challenge 5), with both FedAtt and FedAvg. The results of this experiment were highly improved compared to the previous model and were successful in disaggregating the energy for multi-state devices.

1.6 Organization

This thesis is organized as follows:

- Chapter 1: Introduction - This chapter gives a general overview of the thesis and the main components.

- Chapter 2: Background and Related Work - This chapter provides brief explanations and mathematical intuitions to help readers comprehend various terms and ideas mentioned in most other chapters.

- Chapter 3: A Federated Learning Model With Short Sequence To Point Mechanism For Smart Home Energy Disaggregation - Based on the online Seq2Point model (Short Seq2Point) in this paper [32], there is a proposed method in this chapter to evaluate the performance of deploying FL in NILM. All the findings of this chapter were published in [25]. The UK-DALE and REFIT datasets are used to test the performance of this framework. The results show that, while preventing any data leakage in this system, the performance is still comparable to the centralized version of this method. Besides, different fractions of clients were explored in the REFIT dataset to show the importance of the number of clients.

- Chapter 4: Exploring New Attention-Based Approach In Federated NILM - Every human on this planet has a unique approach to dealing with their environment. This uniqueness can even be found in appliance usage in their houses. This chapter discusses that different clients in the FL framework have different energy usage patterns. Thus, results produce data with different levels of importance in the network training. Using attention-based FL, it was possible to pay

more attention to the buildings with more valuable information than the others. Moreover, to add more security to the network, different noise levels were added to the local parameters to see how the system could perform. Hyperparameter tuning was also done in this chapter to find the best-suited parameters for FedAtt.

- Chapter 5: Utilizing Variational Auto-Encoder Model And Federated Approaches For NILM - One model alone (Short Seq2Point) was insufficient for gauging FL NILM's efficacy. Combining energy disaggregation and FL with the popular model in image classification and natural language processing, Variational Auto-Encoder, led to a dramatic improvement in performance compared to earlier sections of the thesis.

- Chapter 6: Conclusion and Future Work - This chapter draws a few conclusions based on the findings of the various experiments and systems offered in this research. Additionally, this chapter discusses various alternative ways in which additional research could be conducted in this field.

Chapter 2

Background and Related Work

2.1 Background

This section provides some more information regarding the concepts and models that have been used throughout this research.

2.1.1 Non-Intrusive Load Monitoring

NILM is a method of computing the power consumption of individual appliances from a single meter that monitors the total demand of several appliances. Assume that the total power consumption of a house $A(t)$ is the sum of all the appliances' active power consumption in a household. Note that the mains reading (whole house power consumption) at time t is denoted by $A(t)$. The mains can therefore be represented by the formula below (provided by [11]):

$$A(t) = \sum_{i=1}^{I} B_i(t) + c(t) \tag{2.1}$$

Where $B_i(t)$ denotes the power reading of appliance i at time t, I indicates the total number of appliances, and $c(t)$ is the model noise variable. Generally the noise $c(t)$ is defined as Gaussian with mean 0 and variance σ^2. The goal of this formula is recovering the appliance-wise power consumption from one observation of the total consumed energy which makes it a Single-channel Blind Source Separation problem [11]. Several approaches, such as employing domain knowledge to the model or applying signal processing, can be applied to resolve this problem.

Another method to tackle this Single-channel Blind Source Separation problem is supervised learning. For NILM, many datasets on residential electricity are now available. These datasets contain the total household power and the power consumed by each appliance in the house. Assume that B_t is the

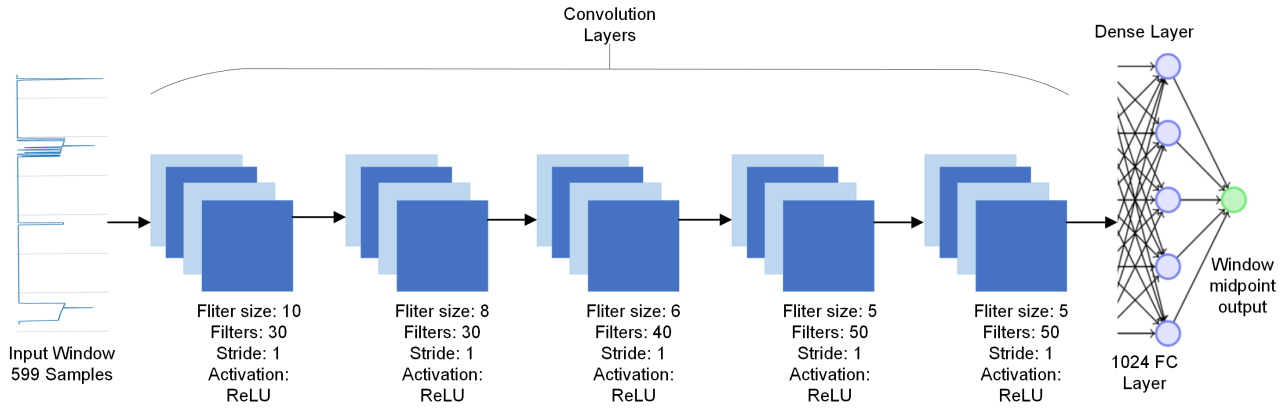

Figure 2.1: The architecture of Sequence-to-Point learning

power of an appliance at time t, and A_t is the total power consumed by the entire house. It is critical to determine a function that maps A to B to solve NILM. The following formula

$$B = G(A) \tag{2.2}$$

is a non-linear regression problem. Several strategies can be used to find function G in the equation 2.2. Convolution layers, which behave like denoising functions and attempt to eliminate the noise part ($c(t)$) in equation 2.1, is one of these methods. Note that the choice of hyperparameters and the number of layers significantly influence the final outcome. Both Sequence-to-Sequence [59] and Sequence-to-Point [68] are solutions for finding the function G. A slightly changed version of the Seq2Point [32] model was chosen for this paper due to its superior performance.

2.1.1.1 Sequence-to-Point Learning

Seq2Point, one of the models for solving NILM, is a data-driven solution that requires the data and the label. Seq2Point learning is based on training a Convolutional Neural Network (CNN), which outputs the midpoint of an appliance given a fixed window of total electric load as input. In the original study [27], which proposed Neural Networks as solutions for NILM, they sampled the power consumption of each appliance every 6 seconds. With a fixed window length of 599 samples, the total duration of a window was 3600 seconds or 1 hour. Even though other articles, such as [11] and [68], used different datasets with other sample rates, they all employed the same window length since they followed the same model for their Neural Network. Figure 2.1 represents the architecture of Seq2Point. The input

of the network is a window of the total electric load $A_{t:t+W-1}$, and the output is the midpoint element b_τ of the target appliance's corresponding window, where $\tau = t + W/2$. The Seq2Point model has the advantage of having a single prediction for each b_t instead of an average of projections for each window. The midpoint element is assumed to be a non-linear function of the mains window in the model. The logic for this assumption is that the status of the appliance's mid element should be related to the information from the mains before and after that midpoint. The Seq2Point architecture constructs a neural network G that maps sliding input window $A_{t:t+W-1}$ to the midpoint b_τ of corresponding output window $B_{t:t+W-1}$. The model is:

$$b_\tau = G(A_{t:t+W-1}, \theta) + \epsilon \tag{2.3}$$

The network parameters are represented by θ, and ϵ is a parameter that compensates the $c(t)$ function. Assuming $\hat{b_\tau}$ as the output of the network, the formula of the loss function is:

$$L = \sum_{t=1}^{T-W+1} (b_\tau - \hat{b_\tau}) \tag{2.4}$$

Given a full input sequence $A = (a_1 ... a_T)$, we first pad the sequence with $W/2$ zeros at the beginning and end to cope with the sequence's endpoints [68].

2.1.1.2 Variational Auto-Encoder

With the assistance of Variational Auto-Encoder (VAE) [29], it is possible to provide a probabilistic description of observation in latent space. As a consequence of this, the encoder will be designed in such a way that it expresses a probability distribution for each latent feature rather than a single value for that feature. This model recently was used as a energy dissaggregation method in [33].

Instead of mapping inputs ($x \in \mathbb{R}^N$) to a single point, the probabilistic framework that the VAE presents distributes them in a distribution over a continuous latent space called z. As a condition of the impossibility of calculating the actual posterior density $p_\theta(z|x)$, the marginal likelihood $p(x)$ cannot be differentiated. In order to solve this problem, the probabilistic encoder $q(z|x)$ was developed to provide an approximation of the actual conditional inference distribution $p_\theta(z|x)$. The variational parameters and the generative model parameters are both learned simultaneously using the variational principle, which states that $\log p(x)$ can be represented using the following equation:

$$\begin{aligned} \log p_\theta(x) &= KL(q_\phi(z|x) \parallel p_\theta(z|x)) + L(\theta, \phi; x) \\ \geq L(\theta, \phi; x) &= \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - KL(q_\phi(z|x) \parallel p_\theta(z)) \end{aligned} \tag{2.5}$$

where $L(\theta, \phi; x)$ represents the variational lower bound that needs to be optimized. The expectation term of 2.5 helps to improve the reconstruction accuracy of the probabilistic decoder $p_\theta(x|z)$, while the Kullback-Leibler (KL) divergence term functions as a regularizer to ensure that the approximate posterior is as close as possible to the prior value $p(z)$. $p(z)$ is assumed to be a centered isotropic multivariate Gaussian with identity covariance $N(z; 0, I)$, and the variational approximation posterior is $N(z; \mu(x), \sigma^2(x)I)$. In general, $p(z)$ is a multivariate Gaussian. Due to the fact that $p(z)$ and $q(z|x)$ are both Gaussian in this scenario, the $KL(q_\phi(z|x) \parallel p_\theta(z))$ equation can be solved using the closed form. In reality, the encoder, $q_\phi(z|x)$, and the decoder, $p_\theta(x|z)$, are both neural networks, with ϕ and θ, respectively, serving as their respective parameters. In this manner, $\mu(x)$ and $\sigma^2(x)$ correspond to the encoder outputs, and they are learned from observed datasets by means of the objective function specified in 2.5.

As the decoder takes its samples from the distribution $z \sim q_\phi(z|x)$, the gradient cannot be back-propagated through the network's stochastic units. To fix this, the reparameterization approach is employed which is given in [29]. The formula$z = \mu + \sigma \odot \epsilon$ is calculated once a random sample of an auxiliary variable $\epsilon \sim N(0, I)$ has been generated. In this context, $\odot$ represents a product of individual elements.

2.1.2 Performance Evaluation Metrics

This thesis uses some well-known evaluation metrics to evaluate different models and frameworks: Accuracy, Recall, F1 Score, Precision, Mean Absolute Error (MAE), and Relative Error in Total Energy (RETE).

2.1.2.1 Regression Metrics

Considering E as the actual consumed energy, $\hat{E}$ as the total predicted energy, T as the length of the input window, b_t as the appliance consumed power and $\hat{b}_t$ as the prediction of consumed power at time t, the definitions of the MAE and RETE are as follows:

$$MAE = \frac{1}{T} \sum_{t=1}^{T} |\hat{b}_t - b_t| \tag{2.6}$$

$$RETE = \frac{|E - \hat{E}|}{max(E, \hat{E})} \tag{2.7}$$

2.1.2.2 Classification Metrics

It is important to note that in many experiments conducted for this thesis related to the amount of energy consumed by an appliance, the appliance is ON only if the amount of power consumed by the appliance is more than a predetermined threshold. In any other case, it is in the OFF state.

Now, before going into details of the following metrics, first the following concepts should be explained:

- *True Positives (TP)* - These are the positively predicted values that were correct, indicating that both the actual value of the appliance's state and the predicted state for the appliance are ON.
- *True Negatives (TN)* - These are the negative numbers that were accurately predicted, and they indicate that the appliance is currently in the OFF state, which is also the state that was predicted for it.
- *False Positives (FP)* - When the actual state is OFF, but the predicted state is ON.
- *False Negatives (FN)* - When the actual state is ON, but the predicted state is OFF.

Accuracy is a performance metric that makes the most sense to most people because it is a ratio of correctly anticipated states to the complete set of states that the appliance can be in.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.8}$$

Precision is the ratio of correctly predicted ON states to the total predicted ON states.

$$Precision = \frac{TP}{TP + FP} \tag{2.9}$$

Recall (sensitivity) is the ratio of correctly predicted ON states to all predicted values that their actual state is ON.

$$Recall = \frac{TP}{TP + FN} \tag{2.10}$$

The F1 score is calculated by taking the weighted average of Precision and Recall. As a result, this score takes into consideration the possibility of both FPs and FNs. Although F1 is more difficult to grasp intuitively than accuracy, it is often more useful in practice, especially when dealing with an uneven distribution of classes.

$$F1 = \frac{2 * (Recall * Precision)}{Recall + Precision} \tag{2.11}$$

2.1.3 Federated Learning

Federated Learning (FL) is a novel machine learning approach that Google presented for auto-correction in mobile keyboards [20, 50]. Also, FL has been put into production use by Apple [47] company, particularly for mobile application development. Data communication between devices (or clients) and the server has always been time-consuming. As a result, if there is a model on the central server, training takes a long time because it requires all data gathered in one location. Furthermore, transmitting data from the client to the server may jeopardize the client's privacy.

Nowadays, several learning systems are working based on offline trained models. Since it is offline, the training model updates weekly or daily depending on the servers. Besides, a considerable amount of data is generated on devices, and sending all the data to cloud-based servers might be impractical. Moreover, data privacy and security requirements are essential concepts that need to be considered. With the help of FL, we can address these problems. The purpose of FL is to model various decentralized devices or servers, keep local samples without data exchange, and ensure data privacy, security, and exclusivity. The FL framework has not been explored enough, especially in NILM, making this combination a state-of-the-art privacy-preserving approach in energy disaggregation. Figure 2.2 shows the architecture of FL. The number of clients can change during this process [16, 70].

Instead of putting all the data on one server and then making a training model, this method allows local models to be made on each device. The global model (the one on the server) is updated with an aggregated form of local parameters. This way, the performance metrics are better cross-validated, and the whole process is more secure because no data is transferred, which is an ethical advantage for this model. The only parameters that will be transferred during this process are the weights of local models. Then, on the central server, there would be an aggregation of all local parameters for validating the global model and sending the aggregated parameters back to all the local models for the next epoch. Two aggregation methods are used in this thesis, which are explained in the following sections.

2.1.3.1 Federated Averaging

Federated Averaging is a way of aggregating all the weights in the local models and building a new global model based on the final average, as suggested in this article [20]. A subset K of clients receive global parameters w_t, during training round t. Clients can start from the same global parameters that has been randomly initialized in the case of $t = 0$. Each client taking part in a round has a local dataset of n_k samples, where k is the index of clients participating. The value of n_k varies depending

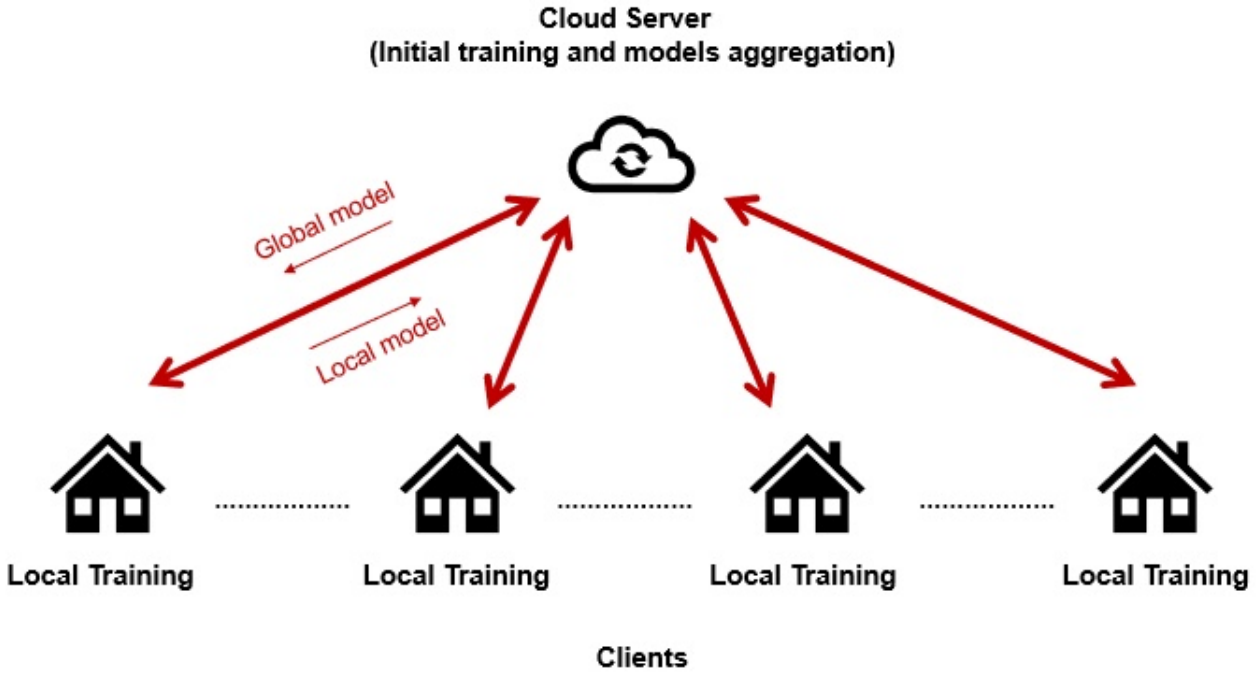

Figure 2.2: Federated Learning architecture in Non-Intrusive Load Monitoring

on the client. In NILM, the n_k of each client is the total number of samples of that specific appliance in that house. The local parameters will be updated according to the desired optimizer used in the local models and the number of epochs. Each client computes the average gradient, g_k, on its local data with the current parameters, w_t. The local model update, ${w^k}_{t+1}$, for a learning rate η, is generated with (provided by [20]):

$$ {w^k}_{t+1} \longleftarrow w_t - \eta g_k. \tag{2.12} $$

After that, the global model performs a weighted aggregate on parameters of the local models to produce new global parameters, w_{t+1} (provided by [20]):

$$ w_{t+1} \longleftarrow \sum_{k=1}^{K} \frac{n_k}{N} ({w^k}_{t+1}), \tag{2.13} $$

Where N refers to the total number of samples of a single appliance in all households.

2.1.3.2 Federated Attention-based

One of the essential aspects of FL is the federated optimization performed on the server side by aggregating the parameters of local models. According to the findings of [23], an innovative federated optimization method called attention-based aggregation, also known as FedAtt, can be implemented in FL to learn from decentralized local models. This research [23], was applied on natural language processing datasets; however here the approach is different. This thesis uses this technique which presents the attention mechanism for federated aggregation in the context of NILM by aggregating the layer-wise contribution of neural disaggregation models of selected clients to the global model stored in the central server.

The technique behind layer-wise FedAtt is depicted in figure 2.3. It is important to keep in mind that this figure only illustrates the first step of the attention-based weight aggregation process. The attention-based FL will be carried out on the global model, which can be found on the right side of the figure. The local models can be found on the left side of the figure. A neural network is incorporated into each one of the local models. In addition, all of the models that are considered to be local models are identical to one another. During the training phase, each local model will acquire a different layer-wise weight compared to the others due to the fact that they each have their own dataset. During the FedAtt procedure, the operations that will be performed on the parameters of the network are represented in this figure by the symbols "-" and "+" respectively. This block will be utilized for every update of the weights in the global model that is performed.

According to 2.3 the definition of each parameter is as follows:

- θ_t: The parameters of global model at time t.
- θ_{t+1}^m: The parameters of m-th client at time $t+1$.
- α_m: The attention-based weights of m-th client.
- w_m^L: The parameters of L-th layer in m-th client.
- w^L: The parameters of global model in L-th layer.

Considering the $L(.,.)$ as the distance function between two time steps, the objective function for optimization in FedAtt is:

$$\arg\min_{\theta_{t+1}} \sum_{k=1}^{m} [\frac{1}{2}\alpha_k L(\theta_t, \theta_{t+1}^k)^2] \tag{2.14}$$

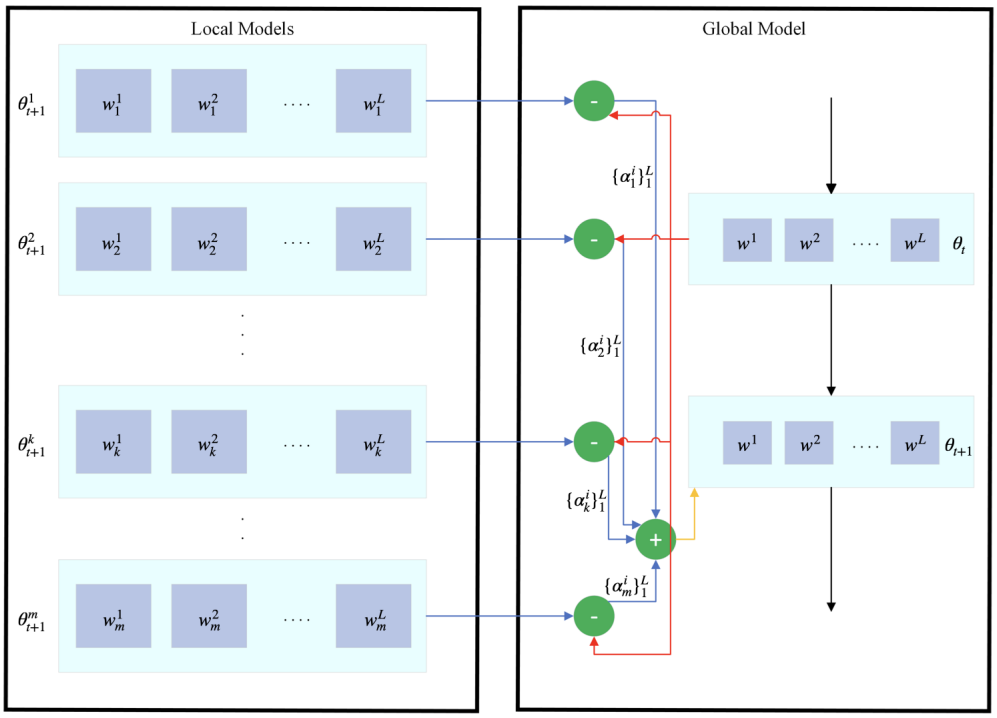

Figure 2.3: A single step presentation of Federated Attention-based in Federated Learning

The purpose of this equation 2.14, is to locate an optimal global model that is comparable to the local models in terms of parameter space, meanwhile taking into account the importance of certain local models at the time of aggregation. Besides, the aim is to use a set of self-adaptive scores as weights to find the minimum possible weighted distance between the global and local models.

In order to calculate the attention-based weight of k-th client in layer l the following formula is used:

$$\alpha^l_k = softmax(\parallel w^l - w^l_k \parallel_p) \tag{2.15}$$

In the $softmax$ function of 2.15, there is a similarity calculation between parameters of l-th layer of the global model and the l-th layer parameters of the k-th local model in the norm of p. Then the $softmax$ function will be applied to calculate the attention-based weight of the k-th client in l-th layer. For each client, there is an array called attention score which contains the attention-based weight of each layer as $\alpha_k = \{\alpha^1_k, \alpha^2_k, ..., \alpha^m_k\}$.

$$\theta_{t+1} \longleftarrow \theta_t - \epsilon \sum_{k=1}^{m} \alpha_k (\theta_t - \theta^k_{t+1}) \tag{2.16}$$

Considering the number of clients as m, to update the global model parameters for the next step with gradient descent, the equation 2.16 is used. ϵ here is the step size.

2.2 Related Work

This section contains several pieces of research that worked on energy management. Also, it presents comparable approaches to our proposed methods for evaluating the performance of a distributed framework for NILM applications. Proposing new models for NILM, experimenting with different training and testing ratios, and adding new tasks and models to generalize the framework for most of the NILM datasets while taking into account the behavioral differences in power consumption of each client are some of the cases that have been studied in various articles.

2.2.1 Federated Averaging

In [34], a FL framework is implemented with the Seq2Point model for NILM. REFIT, Reference Energy Disaggregation Dataset (REDD), and UK-DALE, the most popular datasets in this area, have been used to evaluate the performance of the proposed method. Similar to all the papers trying to secure the NILM system using FL [12, 35, 63], the aggregation method in this article is FedAvg. To get a better generalization, first, REFIT dataset with 20 houses is used for the training. To analyze the performance of FL, common appliances for NILM, including kettle, microwave, fridge, dishwasher, and washing machine, are picked from REFIT data. Taking into account the variations between domains, such as US and UK, this paper tests the REDD dataset with the pre-trained model of UK-DALE for Transfer Learning to investigate FL performance enhancement. Although the results of the FL version model slightly decreased in both cases compared to its centralized counterpart, ensuring the clients' privacy is still a significant achievement. [34] also mentions that examining and considering the communication cost and model efficiency is part of its future goals.

Another important research conducted in NILM and FL is [12]. One of the challenges with NILM is that the amount of labeled data generated by a single household is limited, and the training size dramatically impacts the effectiveness of the deep learning model. Besides, every client's lifestyle differs from each other, making it hard to achieve a generalized model for NILM. [12] investigates communication costs and model accuracy, and for simplicity, it considers that there are just two states for each appliance (ON/OFF). An encoder, a temporal pooling module, and a decoder are used for the disaggregation model while having input signals of 126 samples (12.6 minutes) from the UK-DALE dataset. The chosen appliances are the fridge, dishwasher, and washing machine from houses 1, 2, and 5. The authors propose two parts to their experiments: testing on seen and unseen data. This study compares

the FedNILM results with a centralized version of their model. Then they compare these two models with other novel models from different papers to prove their superiority. Unseen data helps the model distinguish between the same general characteristics of different appliances. Although [12] works on testing seen data, in real-world scenarios, most testing will be done on unseen data.

FedAvg also has been applied to image and text datasets. The authors in [39] present an approach for FL of deep networks based on iterative model averaging (FedAvg). They test it with five different model architectures and four different datasets. Experiments demonstrate that the method holds up well against non-IID data distributions and imbalanced datasets. Compared to synchronized stochastic gradient descent, they can cut the number of communication rounds needed by 10–100x. This work aims to use more computing to reduce the number of communication rounds required for model training. The standard model that the authors adopt is adequate to meet the requirements because evaluating the optimization strategy is the objective in this approach.

The emphasis of this research [39] is on mobile computing devices such as smartphones and tablets because these are the primary computers used by many people [2, 49]. The paper was inspired by the idea that high-quality models can significantly improve the practicality of mobile devices in applications such as image classification and language modeling. To fully explore the FedAvg algorithm's hyperparameters, the authors selected proxy datasets of suitable size for each task. Authors investigate two different approaches to MNIST data partitioning over clients: (I) IID, in which the data is randomly shuffled and partitioned into 100 clients, with each client receiving 600 examples, and (II) Non-IID, in which the data is first sorted by digit label, divided into 200 shards of size 300, and assigned to each of 100 clients. Because most customers will only have two-digit examples, this is a pathological non-IID data partition. Using this method, they could test how much their algorithms deviate from IID. FedAvg trains high-quality models using relatively few rounds of communication, as shown by results on a MLP, two CNNs, a two-layer character LSTM, and a large-scale word-level LSTM.

Although FedAvg, has many real-world privacy benefits, it would be interesting to see more work done in this area to provide stronger guarantees, such as secure multi-layer computation [19] and differential privacy [17, 1, 13, 15] or a combination of these two.

To compare the approaches used in [12, 35, 63, 34] for combining NILM and FL, the table 2.1 has been created.

Paper	DFNILM[34]	FederatedNILM[12]	FedTask[35]	FLNILM[63]
Dataset	REDD UK-DALE REFIT	UK-DALE	Pecan Street [46]	REFIT
Appliance	K, M, F, DW, WM	F, DW, WM	AC, EV, D, O	M, WM, K, DW, TD
Window	599	126	120	599
Metric	MAE	Acc, Pr, Re, F1	Acc, F1, MAE, SAE	F1
Performance	MAE decreased	Better results than S2P, Comparable results to centralized	Better results than centralized in FedMeta	Comparable results to centralized

Table 2.1: Comparing different approaches of investigated articles on Non-Intrusive Load Monitoring and Federated Learning (S2P: Seq2Point, F: Fridge, K: Kettle, M: Microwave, Dishwasher: DW, WM: Washing Machine, AC: Air Compressor, EV: Electric Vehicle, D: Dryer, O: Oven, Acc: Accuracy, Pr: Precision, Re: Recall, SAE: Signal Aggregate Error)

2.2.2 Meta Learning

This paper [35], proposes a comparison between FedAvg, locally trained models, centrally trained models, and FedMeta. Fed Meta employs a nested combination of Meta Learning and FL. Meta Learning can aid in rapidly adapting learned models to the new distribution of NILM datasets. This study aims to accomplish fast decentralized algorithm training and accurate task-adaptive localized energy disaggregation, as motivated by [24, 8]. Another purpose of this research is to improve the accuracy of signal detection. There are undeniable variances in clients' behavior, and a single model may not be the best fit for everyone. In Meta Learning, it is possible to adjust the models locally and train a set of learnable models instead of training a pre-trained model. The chosen dataset for this article is 25 houses from Pecan Street in Austin, and the appliances are electric vehicle, air compressor, dryer, and oven. The choice of appliances here differs from all the other papers working on this subject [34, 12, 63]. For training, it is mentioned that a combination of two houses is used and it has been divided evenly based on the number of clients. Compared to a real-world application of FL, the clients cannot have similar data, which [35] does not follow this rule, jeopardizing the privacy concern. With 120 samples resulting in 2 hours of the input signal, the GRU model is chosen to perform the energy disaggregation on each appliance. [35] concludes that FedMeta performs better than FedAvg and centrally trained models because it fine-tunes a specific model for every new task and shows consistent performance for all four appliances. The article considers investigating fast converging schemes for learning load disaggregation in a federated and distributed fashion for future work.

2.2.3 Differential Privacy in Federated Learning

When a model is traditionally trained, its parameters carry information about the training data. To solve this issue, [1] proposed the concept of Differential Privacy (DP) [14] for learning algorithms. The purpose is to keep a trained model from revealing whether or not a specific data point was used during training. The use of FL makes the learning process more secure. A reliable server will aggregate the decentralized parameters from several clients. The global model's parameters are shared with all the clients, forming a joint representative model without sharing data. Issues regarding security and privacy and the costs associated with communication are significant in centralized learning. FL is a helpful approach for solving these problems. Still, differential attacks can originate from any party involved in federated optimization. A distributed model attack can show the training contribution and dataset of a client. To tackle this problem, this paper [17] proposes a client-sided differential privacy-preserving federated optimization algorithm. In this algorithm, clients' contributions are hidden during

training to balance privacy loss with model performance. Given a large number of participating clients, the proposed method in [17] can maintain client-level DP at a minor cost in model performance.

One of the biggest obstacles to achieving DP in FL is DP noise's effect on model accuracy, especially for deep-learning models with many parameters. Keeping model precision high while providing DP guarantees at the client level is the goal of Federated Sparsified Model Perturbation (Fed-SMP), a novel differentially private FL scheme developed in this article [22]. Fed-SMP, is a technique that involves first sparsifying local models and then perturbing them with additive Gaussian noise [22]. The experimental findings presented in [22] have demonstrated that Fed-SMP, which utilizes two distinct sparsification strategies, can simultaneously improve the privacy-accuracy trade-off and communication efficiency compared to the currently used methods.

Employing central Differential Privacy in FL can provide a good balance between the user's privacy and the model's utility, but doing so necessitates using a trusted server. Using local DP for FL does not require a trusted server; however, this method's privacy-utility trade-off is relatively poor. The recently proposed shuffle Differential Privacy based Federated Learning (DP-FL) [18] has the potential to bridge the gap between central DP-FL and local DP-FL without the need for a trusted server. Despite this possibility, there is still a utility gap when many model parameters are involved. This paper [26] proposes Oblivious and Differentially Private Federated Learning on Trusted Execution Environment (OLIVE). This system leverages a trusted execution environment to combine the benefits of both central DP-FL and local DP-FL into a single solution.

To show that the index information in the gradients is adequate to cause a privacy leakage in the training data, the authors [26] design an attack by an aggregation server. The conceptual framework for the planned attack relies on the hypothesis that the absolute values of the converged gradients of the model parameters have a positive correlation with the training data labels. The attacker, who has control of the aggregation server, obtains index information via a memory access pattern leakage and uses it to estimate the private label information of the training data that the targeted client possesses. In response to these threats, the authors propose a fully-oblivious but efficient algorithm that maintains uniform and secure memory access patterns, thereby protecting users' privacy.

2.2.4 Attention-Based Energy Disaggregation

To solve the problem of NILM, the authors of this paper [48] propose using a deep neural network with layers for regression and classification. The architecture's generalization capability was improved by including an encoder-decoder in the regression subnetwork equipped with a specialized attention

mechanism. The attention mechanism was conceptualized based on the productive applications of temporal attention in fields such as neural machine translation, text summarization, and speech recognition. The proposed deep neural network outperforms the state-of-the-art in all of the experimental conditions taken into consideration, as shown by tests conducted on two datasets that are available to the public (REDD and UK-DALE). They also show that by using attention modeling, the network can accurately detect when an appliance is turned ON or OFF and locate signal sections with high power consumption, which are very interested in energy disaggregation.

The use of deep neural networks as the disaggregation model leads to an increase in the amount of computational complexity. To find a solution to this issue, the researchers in [62] included attention processes in their neural networks. Dot attention and additive attention were both tested in this research. The results of the experiments demonstrated that the proposed architecture achieves faster or equivalent training and inference time with just a slight loss in performance dependent on the device or the dataset.

2.2.5 Behavioral Analytics Energy Consumption and Forecasting Electrical Usage

For the modern power system to function correctly, consumers must engage in energy usage that is efficient, responsible, and considerate of the environment. Besides, utility companies are now trying to develop demand-side management and demand response to cut costs and enhance profits. Thus, analyzing, forecasting, and visualizing energy time-series data is important in understanding how people use energy over time. As a result of these patterns, it is possible to deduce and assess how consumers' energy consumption habits and forecasting trends are influenced by their appliance-appliance relationships in a home in terms of the time of day, weekday, month, and season of the year. [55].

One of the most critical challenges that exist is the fact that there are numerous links between appliance consumption and concurrent data streams. The authors of this paper [55] propose clustering, frequent unsupervised incremental mining analysis on energy time-series, and Bayesian network forecasting as some solutions to the mentioned problems. They use Support Vector Machines (SVM) and Multi-Layer Perceptron (MLP) as their models because they are two well-known methodologies for classification approaches.

The authors of [55] used time-series energy consumption data from UK-DALE [28], AMPds2 [37], and a synthetic dataset for training. The evaluation results support the human behavior-energy consumption hypothesis. This is shown by appliance-to-appliance and appliance-to-time correlations. The authors noticed that appliances such as laptop, monitor, and speakers manifest associations. During

incremental mining, associations between these appliances increase, and new appliances such as washing machines, kettle, and running machines form. The authors can deduce the occupiers' behavioral preferences from these linkages, such as "occupants like to work on the computer and/or listen to music while washing clothes" and "work on the computer and/or workout while cooking."

Another critical issue is recognizing the appliances that spend the most electricity and extracting sophisticated interdependencies across many appliances simultaneously. Because energy consumption data is constantly being generated, appliance associations can change. The authors of this study [56], offer an unsupervised progressive incremental data mining mechanism for mining smart meter energy usage data for frequent patterns. This has the potential to promote end-user participation and energy demand management. This paper offers the findings of an evaluation of the proposed process using real-world smart meter data.

This article [56], tested intermediate and final results using incremental frequent pattern mining on data regarding the energy use of five UK-DALE households [28], as well as a synthetic dataset. This study lends credence to the theory that human actions impact the home's energy use patterns. These patterns are teachable by connecting appliances to one another. The authors demonstrate that customer behavior influences household energy use using three different households. Energy consumption curves complement the finding of patterns and rules.

Finding irregular cases affecting power system planning is yet another component of examining customers' behavior concerning the energy consumed. The purpose of this work [66], is to present a case study that mines trends in residential electricity usage and identifies abnormal users through hierarchical clustering. A process model and the various techniques for mining electricity usage patterns in a smart grid will be presented after a brief discussion on hierarchical clustering. This paper proposes a case study that is derived from the records of the daily electricity usage made by 300 people of Kunshan, China, from the 16th of November to the 16th of December of 2014. Their experiments show that most residents in Kunshan have a similar power consumption pattern, and their consumption trend was strongly connected with temperature variations. Although the number of people with unusual electricity usage patterns is small, they must be noticed. Their electricity consumption patterns are critical for power system planning, operation, decision-making, and strategy formulation for Demand Side Management (DSM) [66].

In addition, providing a framework for mining typical load profiles in buildings to drive energy management techniques is an important activity that can contribute to creating profitable smart cities.

Building load profiles are broadly described using the most recent scientific material in [7]. To provide a clear picture of how much energy is being used at various scales and levels, the authors combined a variety of pattern recognition and classification algorithm.

Another unique methodology for estimating building electrical usage is cite's hybrid model, which combines a clustering algorithm and the Autoregressive Integrated Moving Average (ARIMA) proposed by [44]. This method for forecasting the electrical peak load of university buildings involves clustering data for a year, including the forecasting day, using K-means clustering, and utilizing the result to forecast the electricity peak load of university buildings. If electric peak load could be predicted accurately a few hours before peak hours, management might have enough time to devise ways to reduce peak load. This strategy can also be used in demand response to lower electricity bills by avoiding electricity use during peak hours [44].

The hourly statistics of power load for the East Campus of Chubu University in 2017 and 2018 are used for this paper [44]. A Building Energy Management System (BEMS) at Chubu University measures energy loads at each building's transformer and stores the results in the BEMS server every minute. Hourly data is derived by adding the minute data together, and there are 8760 data points for each building in a year's worth of data. The authors demonstrated that their innovative idea of gathering days with electrical demand patterns comparable to forecasting a given day provides more accurate results than the ARIMA model does [44].

Besides the previous techniques to deal with different power consumption patterns of the users, this study [9] presents a data-driven approach to aggregating residential customers based on the customers' power usage habits over a year. They implement their proposed approach using the smart meter dataset provided by the smart metering project by the Commission for Energy Regulation (CER) of Ireland [3]. The researchers begin by grouping each client's daily load curves into similar patterns. Next, they calculate the average load curves across all clusters. The cluster analysis performed on the average load curves demonstrates the existence of four separate classes. The behavior patterns of residential customers can be characterized by first computing the percentages of each average load curve group during various seasons and then proceeding to do so in a subsequent step. A hierarchical clustering method is utilized to aggregate and classify customer data.

Peak power demand is becoming an increasingly major challenge for UK networks because it produces imbalances between demand and supply, which in turn has a detrimental impact not only on the expenses of the system but also on the environment [61]. This article [61] proposes methods for

combining time-use data with metered data to reveal peak electricity demand behaviors and sequences. It develops new measures like entropy indexes and occupancy variance to measure synchronization and active occupancy. Sequence analysis, optimal matching, and grouping improve time usage studies and allow comparisons across end-user socioeconomic categories.

2.2.6 Summary of Related Works

To organize all the approaches investigated in the related work section, the following tables 2.2, 2.3, 2.4, 2.5, 2.6 and 2.7 have been created. A summary of all the models and methods in this thesis is shown in table 2.8 to compare them with the papers in the related work.

Group	Federated Learning				
Paper	DFNILM[34]	FederatedNILM[12]	FedTask[35]	FLNILM[63]	ComEfficient[39]
Method	FedAvg	FedAvg	FedAvg	FedAvg	FedAvg
	TL	-	FedMeta	-	-
Model	Seq2Point	Enc+Pool+Dec[69]	GRU[10]	Seq2Point	MLP
	-	-		-	CNN
	-	-		-	LSTM
Dataset	UK-DALE	UK-DALE	Pecan St[46]	REFIT	MNIST
	REDD	-		-	Shakespeare
	REFIT	-		-	CIFAR-10[31]

Table 2.2: All the papers related to Federated Learning

Group	Differential Privacy in Federated Learning	
Paper	DPFLClient[17]	Fed-SMP[22]
Method	FedAvg	FedAvg
DP	-ClientDP -Adding Gaussian distortion to the sum of all updates	-ClientDP -Unbiased random sparsification -Biased top-k sparsification
Dataset	MNIST	MNIST
	-	Shakespeare
	-	SVHN

Table 2.3: Papers related to Differential Privacy in Federated Learning

Group	Differential Privacy in Federated Learning	
Paper	CLDP-SGD[35]	OLIVE[26]
Method	FedAvg	FedAvg
DP	-Shuffling the received gradients then sending them to the server	-Having the utility of centralized DP-FL -No need for a trusted server -Preventing privacy risks by memory access patterns
Dataset	MNIST	MNIST
	-	CIFAR10, CIFAR100
	-	Purchase100

Table 2.4: Papers related to Differential Privacy in Federated Learning

Group	Attention-Based Energy Disaggregation	
Paper	AttNILM[48]	SAED[62]
Model	Regression subnetwork: Encoder– CNN + RNN + Bidirectional LSTM Attention– Single layer feed forward Decoder– Fully connected layers Classification subnetwork: Seq2Seq[68]	Inspired by Window GRU[32] Except one Bidirectional GRU is replaced by Attention layer (Multiplicative/Dot)
Dataset	REDD	REDD
	UK-DALE	UK-DALE
	-	REFIT

Table 2.5: All the papers related to attention mechanism for Non-Intrusive Load Monitoring

Group	Behavioral Analytics and Energy Consumption Forecasting		
Paper	BAEnergy[55]	MininigBP[56]	HouseEP[66]
Goal	-Analyzing and forecasting energy usage	-Extracting complex interdependencies among appliances -Identifying appliances with high energy usage	-Identifying abnormal users -Mining electricity usage patterns
Method	-Incremental frequent pattern mining -Clustering analysis -Association rules extraction	-Unsupervised progressive incremental data mining	-Hierarchical clustering -Classification and Regression Tree
Dataset	Synthetic dataset	Synthetic dataset	Kunshan City
	UK-DALE	UK-DALE	-
	AMPds2	REFIT	-

Table 2.6: Papers on energy forecasting and behavioral analytics

Group	Behavioral Analytics and Energy Consumption Forecasting		
Paper	MLEM[7]	ARIMA[44]	DABP[9]
Goal	-Mining electricity usage patterns to support advanced energy diagnosis	-forecasting electricity peak load	-Aggregating residential customers' power usage based their pattern -Identifying critical seasonal characteristics
Method	-Hierarchical clustering -CART	-Clustering technique -ARIMA model	-Hierarchical clustering -K-means
Dataset	Heating/cooling mechanical room Politecnico di Torino	University building in Japan	Commision for Energy Regulation of Ireland

Table 2.7: Papers on energy forecasting and behavioral analytics

Thesis			
Method	FedAvg	FedAtt	FedAvg FedAtt
Model	Short Seq2Point	Short Seq2Point	VAE
Appliance	WM, DW, M, F	WM, DW, M, F	K, WM, DW, M, F
Window	5-20 mins	5-20 mins	1024 samples
Metric	Acc, F1, Pr, Re, MAE, RETE	Acc, F1, Pr, Re, MAE, RETE	Pr, Re, F1, MAE
DP	-	-ClientDP -Adding scaled Gaussian distortion to client's parameters	-ClientDP -Adding scaled Gaussian distortion to client's parameters
Dataset	UK-DALE REFIT	UK-DALE REFIT	UK-DALE

Table 2.8: Proposed approaches for Non-Intrusive Load Monitoring and distributed learning in this thesis (F: Fridge, K: Kettle, M: Microwave, Dishwasher: DW, WM: Washing Machine, Acc: Accuracy, Pr: Precision, Re: Recall)

Chapter 3

A Federated Learning Model with Short Sequence-to-Point for Smart Home Energy Disaggregation

3.1 Introduction

Some of the popular methods for energy disaggregation are Sequence-to-Point (Seq2Point) learning [68] and Sequence-to-Sequence (Seq2Seq) learning. The authors of this paper [59], first proposed Seq2Seq learning on text data. For this network, they used a Long-Short Term Memory (LSTM) to map the input sequence to a vector of fixed dimensionality and then another deep LSTM to decode the target sequence from the vector. After that, the authors of [27] combined RNN and CNN for NILM. In Seq2Seq, a sliding window of main input power is mapped to a corresponding window of the output appliance power[68]. Still, dealing with lengthy input sliding windows is difficult in this technique, resulting in a merging prediction of separate windows. Although Seq2Point learning improved NILM considerably, the findings were only tested on the same data domain.

To tackle the issue mentioned above, the research in [53] proposed two Transfer Learning strategies: Appliance Transfer Learning (ATL) and Cross-domain Transfer Learning (CTL). It demonstrated that the dormant attributes of a complicated appliance (ex., washing machine) could be transferred to a simple device such as kettle. With the help of TL in smart homes, the number of installed sensors on each appliance will decrease since it is possible to monitor their consumption using the extracted deep features of other devices. Another suggestion offered in this study was that Seq2Point learning

is transferable. When the training and test data are in the same domain, Seq2Point learning can be applied to the test data without fine-tuning; when the train and test data are in separate domains, fine adjustment is required before applying Seq2Point learning to the test data [53].

Seq2Point and another approach known as Gated Recurrent Units were examined in [32], and the results showed that both function similarly. Seq2Point performs slightly better only in the kettle. They concluded that both of these methods are appropriate for real-time energy disaggregation. In addition, the authors of [4] presented a real-time NILM technique based on a CNN and K-nearest neighbor combination. They used synthetic data rather than a large dataset to test their model on eight different appliances, resulting in a less expensive and faster approach.

Several recent publications, including [51], and [5], investigated novel event-based optimization techniques and compared various machine learning models to determine the optimal model with the least amount of computation and the highest accuracy. Another recent approach in NILM has been published in an article [42] proposing a novel framework for multi-label disaggregation, which they call multi-NILM. Combining a lightweight disaggregation model and the dimensionality reduction method called Signal2vec [43] has shown encouraging results.

Performing the NILM has always been investigated in a centralized fashion, risking the clients' privacy in the system. This chapter which is published in [25], sheds light on the details and results of a distributed learning framework called FL on a Short Seq2Point model for energy disaggregation.

3.2 System Model

The primary goal of this study is to apply the FL framework on energy disaggregation to improve NILM performance while boosting system stability and safety. To put it differently, FL aims to ship models to data without breaching privacy laws. The parameters for the global model are calculated using the original FedAvg [20]. The advantage of adopting FedAvg in this study over other averaging approaches is that it is a less complicated method to apply and guarantees that we will achieve our goals. Using an optimized version of Seq2Point learning, the whole-house power reading is disaggregated into appliance-wise energy usage in each local model. Each appliance in a house has its neural network in the FL framework while considering the houses as clients. The local models for this framework are made for all the similar appliances from different residences. To properly comprehend the proposed architecture of this article, it is required to first investigate the structure of the chosen datasets.

3.2.1 Dataset

The datasets that are in the energy disaggregation area are mostly large. Thus, working with these datasets, such as UK-DALE [28] or REFIT [41], especially while combining with FL, is a time-consuming process. According to most of the papers in this area, [68, 33, 23], it is possible to select a reasonable ratio of the datasets and apply the desired methods to them. In case of good results, the methods can be applied to the whole dataset for better accuracy and precision.

Moreover, Several appliances are found in a single household, and it would be challenging to consider all of these appliances and successfully execute a model for each one of them individually. Consequently, it is preferable to pick the ones that supply superior information regarding power use over an extended period, and most households already have them. It would be much simpler to describe a generalized usage pattern for that particular appliance if it were done in this fashion.

In the following subsections, two different datasets that were deployed for testing the proposed model are described.

3.2.1.1 UK-DALE

The results of this chapter are reported using the UK Domestic Appliance Level Electricity (UK-DALE). From November 2012 to January 2015, every 6 seconds, UK-DALE data records both the electricity usage of the whole house and appliance-wise power consumption of five houses [28]. With roughly half a billion records, the UK Energy Research Centre Energy Data Centre (UKERC-EDC) published this dataset, which is regarded as one of the largest energy time-series datasets [28]. Although the dataset includes measurements for over ten different types of appliances, for this research, microwave, fridge, dishwasher, and washing machine, which are common appliances for NILM algorithm evaluation, are used. Figure 3.1 presents a sample raw data of this dataset. The following table 3.1 shows the buildings that are used for each appliance.

The parameters of the table 3.2 are used to preprocess the data for the Short Seq2Point model for each appliance.

3.2.1.2 REFIT

To further examine the performance of the proposed framework in this research, the new version of the REFIT Electrical Load Measurements dataset, called cleaned REFIT, including aggregate and appliance-level data for 20 houses, is used. The measurements in this dataset were time-stamped and

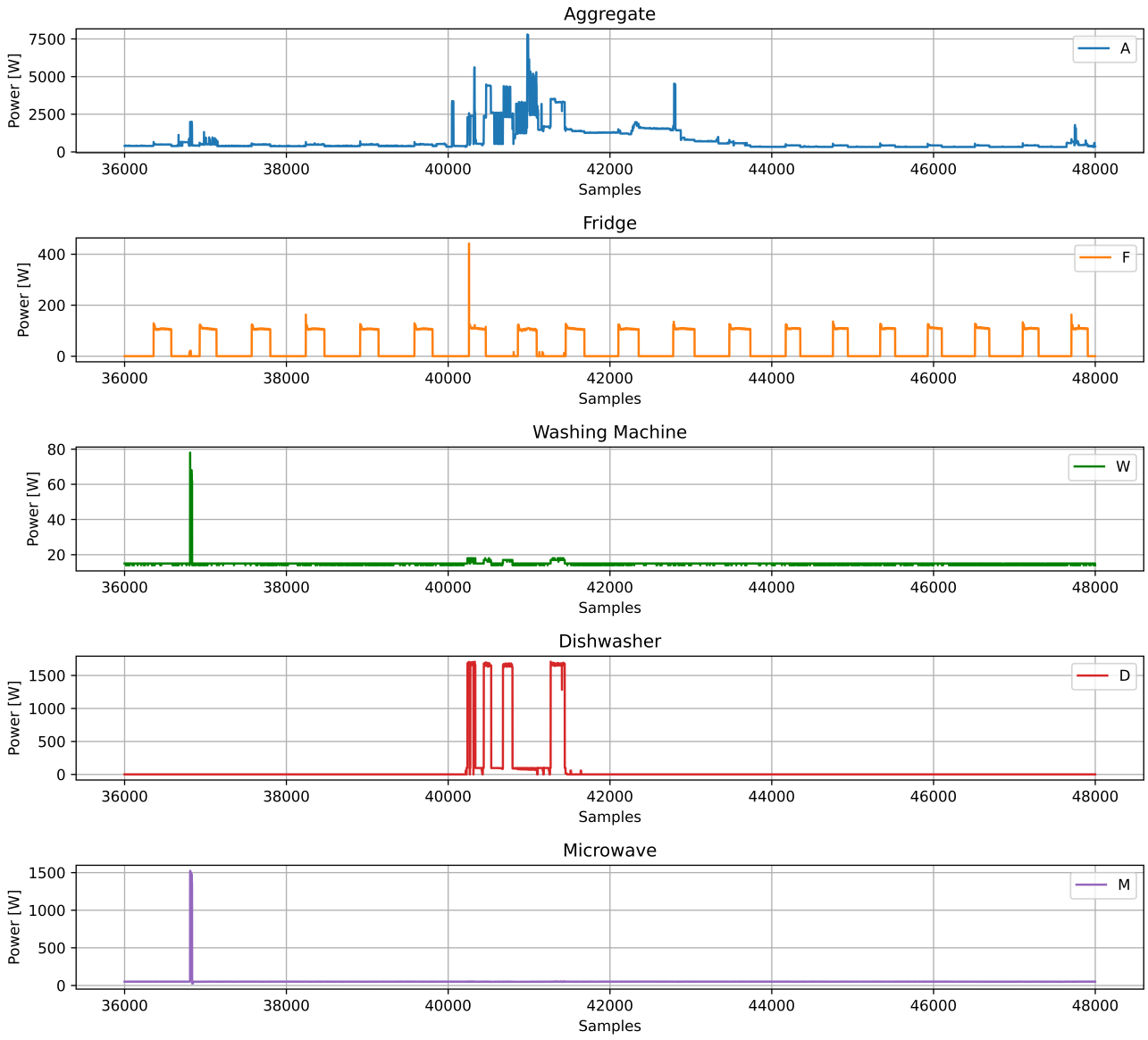

Figure 3.1: A sample of UK-DALE data for chosen appliances and aggregated power from house 5

House	Fridge	Dishwasher	Washing Machine	Microwave
1	√	√	√	√
2	√	√	√	√
3				
4	√			
5	√	√	√	√

Table 3.1: The buildings that are used for each appliance in UK-DALE

Appliance	Fridge	Dishwasher	Washing Machine	Microwave
Maximum	200	3000	2500	3000
Mean	200	700	400	500
On Threshold	50	10	20	200
Std	400	1000	700	800

Table 3.2: Important parameters in appliances in order to preprocess data and recognize state changes (Std: Standard deviation of consumed power)

sampled every 8 seconds. This dataset is intended for use in research about energy conservation and advanced energy services, such as Non-Intrusive Load Monitoring, demand response measures, tailored energy and retrofit advice, appliance usage analysis, consumption and time-use statistics, and smart home and building automation [41].

Same as the UK-DALE dataset, some of the appliances that are more common in NILM application such as fridge freezer, dish washer, washing machine and microwave are chosen from this dataset. The difference here with UK-DALE is the fact that in UK-DALE, the fridge does not have any freezer in it. Figure 3.2 represent a sampled raw data of REFIT.

Moreover, in order to preprocess the dataset, the following parameters in table 3.3 are used for different appliance. After investigating the REFIT dataset with MILMTK package, the buildings in table 3.4 were used for our proposed experiment.

In addition, some buildings do not come equipped with all of the appliances that are considered in the model that has been proposed. Besides, some of the meters were measuring the total combined power of two or more appliances. The houses that were selected for the appliances are the ones that have

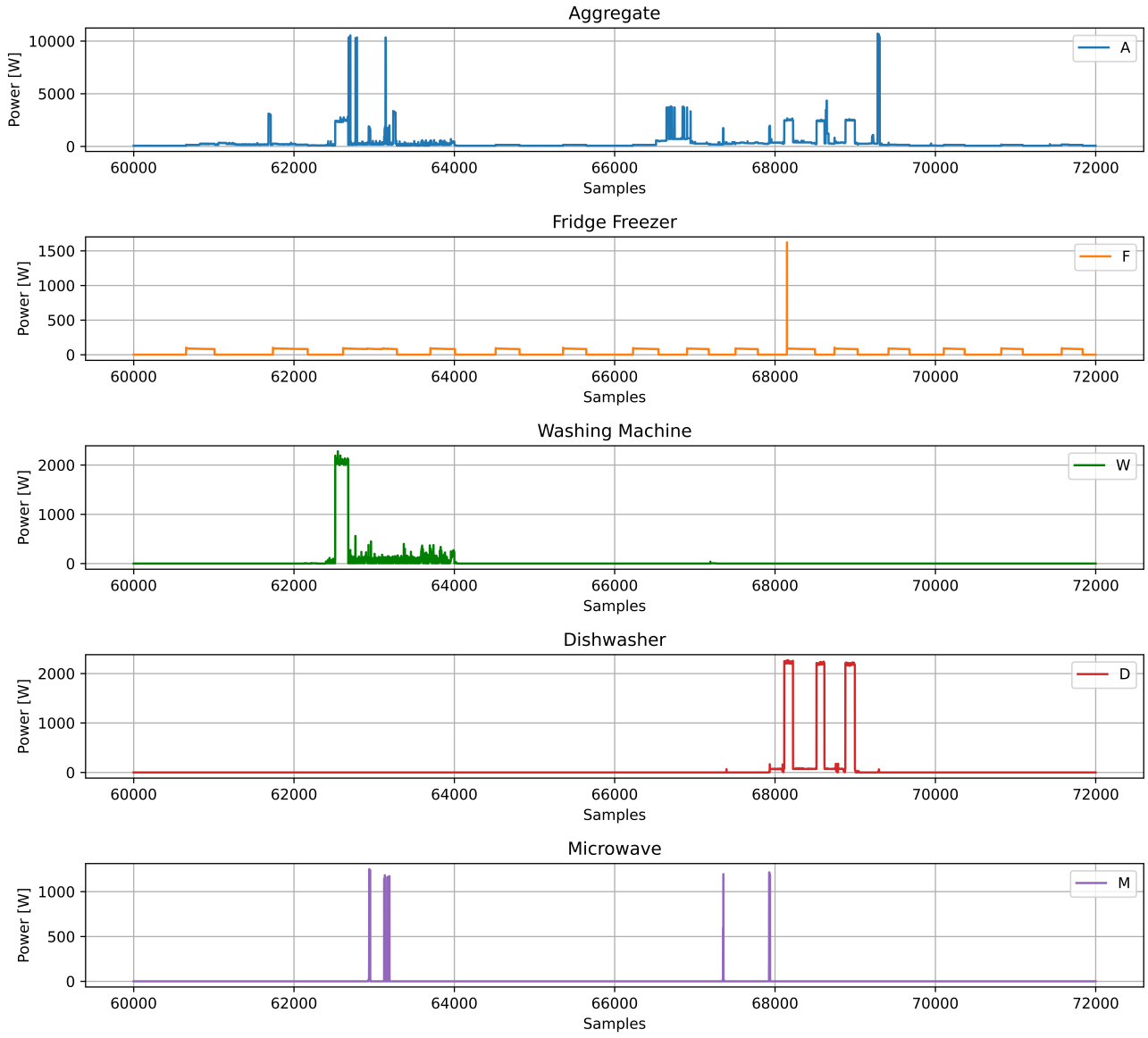

Figure 3.2: A sample of REFIT data in house 2

individual meters that are isolated for each appliance. This was done so that the final results would be accurate. Table 3.4 provides more detailed information about the dataset and selected buildings. The buildings that are not selected for each appliance either do not have that specific appliance or there have two same appliances.

3.2.2 Further Details of Federated Short Sequence-to-Point

Before the FL algorithm begins, the number of clients must be specified. Since data in each client of FL must be isolated, it is not acceptable to use data from local models in the global model for testing

Appliance	Fridge	Dish Washer	Washing Machine	Microwave
Maximum	3969	3968	3968	3778
Mean	200	700	400	500
On Threshold	50	10	20	200
Std	400	1000	700	800

Table 3.3: Important parameters of appliances in REFIT dataset in order to preprocess the data and recognize state changes

Appliance	House
Fridge Freezer	2, 5, 6, 9, 12, 13, 15, 16, 19, 21
Dishwasher	1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 15, 16, 18, 20, 21
Washing Machine	2, 3, 5, 6, 7, 10, 11, 13, 15, 16, 17, 19, 20
Microwave	2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 15, 17, 18, 19, 20

Table 3.4: The chosen buildings for each of the appliances in REFIT

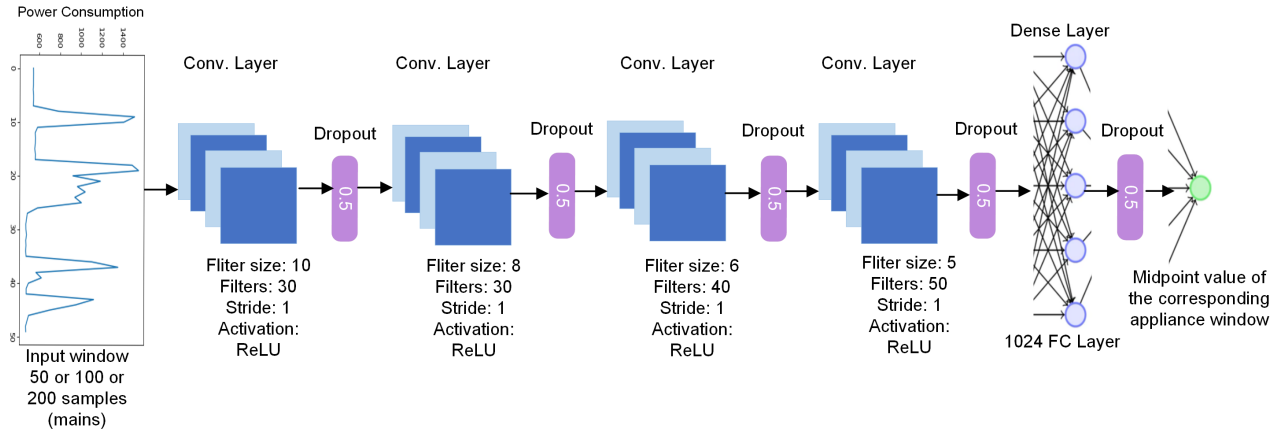

Figure 3.3: The architecture of Short Sequence-to-Point learning

purposes.

The Short Seq2Point is deployed as the CNN model to calculate the appliance-wise power consumption. The big difference between this model and the original Seq2Point is that it has an additional dropout with the value of 0.5 between the layers, and the number of convolution layers is 4 instead of 5

[32]. Figure 3.3 demonstrates the whole architecture of Short Seq2Point. Besides, to make this model compatible with the online disaggregation applications, the duration of the input window is downsized from 1 hour to 10-20 minutes [32].

The following process is repeated four times for the microwave, fridge, dishwasher, and washing machine. In order to start the framework, the following procedure is required:

Step 1: global model is created. (The model is Short Seq2Point and the weights are randomly generated)

Step 2: Local models are generated using the same parameters as global model.

Step 3: Local models are trained for each house and the same appliance based on the chosen number of epochs.

After that the steps shown below, are repeated in a loop:

Step 1: The weights of the global model are extracted.

Step 2: The weight of the local models is set the same as extracted weights of the global model.

Step 3: Local models are trained for a chosen number of epochs.

Step 4: With FedAvg [20], the averages of all weights in local models are calculated.

Step 5: The weights of the global model are updated with the resulting weights of FedAvg.

Step 6: There is no training process in the global model.

Step 7: Using test data, the network's performance will be evaluated according to the performance evaluation metrics in the global model.

When a maximum iteration or a stopping criterion is reached, the above looping process ends.

Assuming w_0 as the initialized weights of global model and local models, *global_epoch* as the number of epochs in the global model, and *client_num* as the total number of clients, algorithm 1 presents the structure of the acrshortfl framework.

3.2.2.1 UK-DALE Preprocessing and Details

In all the appliances except the washing machine, house 5 is used for testing, and the remaining buildings are considered for training data for each local model. For the washing machine, house 2 is used to test

Algorithm 1: FedAvg algorithm

Initialize w_0 `// Randomly initialize the global model`

while *global_epoch* > 0 **do**

  $w_n \leftarrow$ (Weights of global model in this epoch) ;

  for $k \in$ *client_num* **do**

    Create local_model k;

    $w_t^k \leftarrow w_n$;

    $w_{t+1}^k \leftarrow$ (Train local_model with w_t^k for chosen number of epochs);

  end

  $w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{N}(w^k{}_{t+1})$ `// Federated AVG. on local weights`

  Evaluate global model using w_{t+1}

end

the global model. The reason behind this way of data separation is to be able to compare the results to [32]. Consequently, according to the table 3.1, the maximum number of local models is 3 in the fridge, and house 5 is for testing. This value can be changed from 2 to 3 to evaluate the importance of the population in FL. For this study, the fridge and microwave have 50 samples, Dish Washer has 100 samples, and Washing Machine has 200 samples as their input window.

3.2.2.2 REFIT Preprocessing and Details

The main advantage of the Short Seq2Point model is the ability to obtain a real-time solution for NILM. In order to accomplish this, the window size should be decreased so that the duration is between 5 and 20 minutes [32]. It has been shown that there is a trade-off between window size and appliance type [32]. Some appliances might function better under smaller windows, while others require long ones. Since the input windows of dishwasher, fridge freezer, microwave, and washing machine, respectively, are 10, 5, 5, and 20 minutes long [32], it is necessary to convert these durations to the correct number of samples in REFIT. Thus considering the fact that the sample rate is per 8 second in REFIT, the input windows are as: dishwasher (75 samples), refrigerator (38 samples), microwave (38 samples), and washing machine (150 samples).

The training, testing and validation data in REFIT is as follows:

- Fridge Freezer: Train[2, 5, 6, 9, 12, 13, 15, 19], Test[16], Validation[21]

- Dishwasher: Train[1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 15, 16, 18], Test[20], Validation[21]
- Washing Machine: Train[2, 3, 5, 6, 7, 10, 11, 13, 15, 16, 17], Test[19], Validation[20]
- Microwave: Train[2, 3, 5, 6, 8, 9, 10, 11, 12, 15, 17, 18, 19], Test[4], Validation[20]

The REFIT dataset, in contrast to the UK-DALE dataset, contains a higher number of households, which in turn results in an increased number of clients for the FL framework. Random percentages of 50% and 75% were selected for each appliance so that it could be demonstrated how the quantity of clients can have an effect on the system's overall performance.

3.3 Results Analysis

The input window size has been found to impact the model's overall performance. While some appliances are compatible with larger window sizes, others, on the other hand, work with shorter ones [32].

3.3.1 Results for UK-DALE Dataset

Comparing the results of our paper with online NILM [32] in table 3.5 presents that although FL mainly is for adding a privacy layer to the model, it primarily provides comparable results. In the dishwasher, all the metrics are better than the original paper. The situation is the same for the fridge except for MAE, which is slightly different. In microwave, accuracy, precision, F1 score, and MAE are much better. The only appliance that has not been significantly improved compared to the original paper is the washing machine. The reason might be that since the washing machine is a multi-state device, having just 2 clients for the FL model makes it hard to differentiate between its ON and OFF states. In our chosen dataset, since the range of the buildings for training was 2 to 3, it caused less accurate results in some cases. However, the supremacy of FL would be more evident with a higher number of nodes or clients.

3.3.2 Results For REFIT Dataset

Since there was no article on centralized Short Seq2Point on the REFIT dataset, a centralized model has been created for REFIT to compare the results with FedAvg.

As can be seen in the table 3.6, the majority of the results for the various metrics in FedAvg were improved when compared to the centralized model. The value of the F1 score, which is higher in all four appliances compared to the centralized model, is maybe the most noticeable enhancement that can be made. The microwave is the only device whose results have improved across all metrics. Since

	FedAvg Short Seq2Point				**Original Short Seq2Point [32]**			
	F	DW	WM	M	F	DW	WM	M
Accuracy	**0.61**	**0.97**	0.92	**0.98**	0.54	0.96	**0.97**	0.91
Precision	**0.47**	**0.64**	0.13	**0.06**	0.42	0.47	**0.26**	0.01
Recall	**0.8**	**0.53**	**0.75**	0.42	0.74	0.43	0.55	**0.79**
F1 Score	**0.59**	**0.58**	0.22	**0.1**	0.53	0.45	**0.35**	0.03
MAE (Watt)	54	**20**	29	**61**	**51**	21	**17**	103
RETE	0.29	**0.01**	0.56	0.63	0.29	0.07	**0.28**	**0.16**

Table 3.5: Comparison between Federated Short Sequence-to-Point results and the results of online NILM [32] on UK-DALE (F: Fridge, M: Microwave, Dishwasher: DW, WM: Washing Machine)

washing machine is a multi-state appliance and it might be complicated to recognize its states, its well performance in both centralized and distributed methods is notable.

Even though the FedAvg results are encouraging, it is essential to pay attention to the window sizes selected for each appliance. Since the goal of this method is to make NILM run more quickly, it may forego the superior results that can be obtained from a larger window size, just like the previous works that have been published in this field [63]. Because none of the other articles attempted to employ all of the available households for the training in REFIT dataset [63], the results of this experiment 3.6 are more accurate and dependable than those in other articles.

According to the data presented in figure 3.4, the performance of the same measures can potentially improve when the number of customers is increased. Out of 24 experiments shown in 3.4, in 14 cases the performance either increased or remained within the same range. However, in the microwave, it was different. It is possible that the parts of the data that were selected for the training did not contain many ON states for this appliance. As a result, the network may not have been trained properly to predict realistic values. In addition, when FL is used in the real world, hundreds of clients can still utilize the network simultaneously. This provides a more comprehensive dataset for the network to be trained on, which may improve overall performance. Also, choosing proper thresholding for microwave, which is a multi-state appliance can change the results of these experiments.

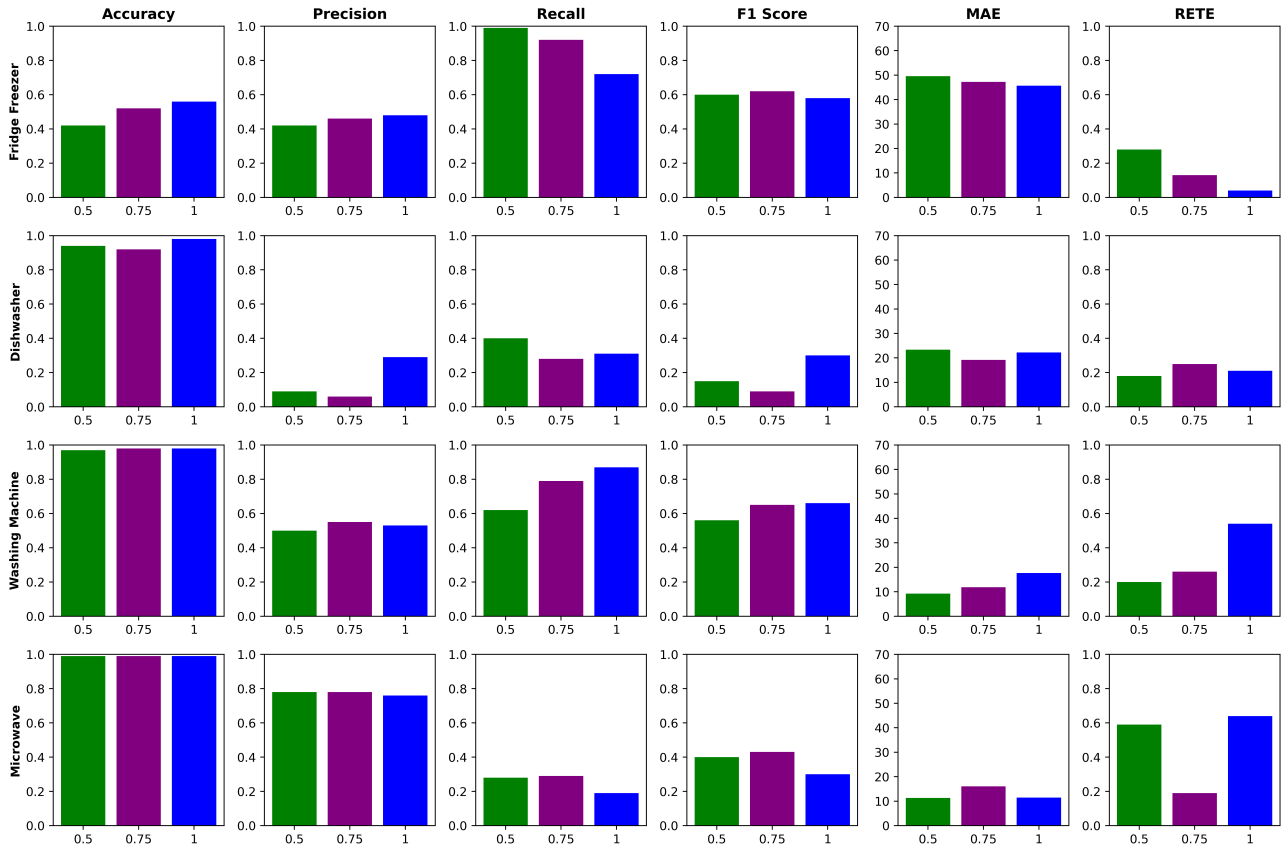

Figure 3.4: Comparison between the performance of different clients' fractions of REFIT

	Centr. Short Seq2Point				**FedAvg Short Seq2Point**			
	FF	DW	WM	M	FF	DW	WM	M
Accuracy	**0.57**	0.94	0.97	0.99	0.56	**0.98**	**0.98**	**0.99**
Precision	**0.5**	0.1	0.45	0.59	0.48	**0.29**	**0.53**	**0.76**
Recall	0.54	**0.49**	**0.93**	0.03	**0.72**	0.31	0.87	**0.19**
F1 Score	0.25	0.18	0.6	0.06	**0.58**	**0.3**	**0.66**	**0.3**
MAE (Watt)	**45.86**	22.95	**8.39**	12.37	45.66	**22.2**	17.65	**11.49**
RETE	0.06	0.54	**0.23**	0.7	**0.04**	**0.21**	0.54	**0.64**

Table 3.6: Comparison between Federated and centralized Short Seq2Point results on REFIT (FF: Fridge Freezer, M: Microwave, Dishwasher: DW, WM: Washing Machine, Centr.: Centralized)

Chapter 4

Deploying New Attention-Based Approach in Federated Non-Intrusive Load Monitoring

4.1 Introduction

Federated Learning was first described in this article [38], which relied solely on average arithmetic operations applied to local models, with respect to the number of samples on each client device serving as the scale factor for the average. Following this aggregation method, none of the recent articles that make use of the FL framework and NILM [34, 12, 35, 63] have investigated any alternatives to the FedAvg technique of aggregation. Consequently, attention-based aggregation will be investigated in this chapter to evaluate the possibility of improving the overall performance of FL.

As discussed earlier, clients of a given energy consumption system could have different power usage patterns for various appliances. For example, some of them might enjoy having coffee first thing in the morning; as a result, they use their coffee maker daily; in contrast, some people might hardly ever turn on this machine. Some people enjoy baking, so they always use their hand mixer. Also, some people might be students who do not have enough time to prepare meals, which leads to an ongoing presence of frozen food in their microwave. Different energy consumers may have different preferences for each appliance. Therefore, transmitting additional information about each client to the central server could significantly impact the system's overall performance.

This extra information can be sent to the server by acquiring an attention mechanism. The attention

mechanism is a vector that assists in orienting perception. It was in the computer vision area when it initially gained widespread recognition. This method was utilized in Recurrent Neural Network models for picture categorization [40]. After that, it was utilized extensively in Seq2Seq NLP applications, such as neural machine translation [6]. The authors of [36] broadened the application of attention-based RNNs and suggested two novel mechanisms: global attention and local attention mechanisms.

Additionally, the attention mechanism may be utilized in CNN models for the encoding of sentences in order to model sentence pairs [67]. In contrast to the extensive research that has been conducted on attention mechanisms in NLP, the magnitude of research that has been conducted in this field with relation to NILM needs to be enhanced, particularly concerning the FL framework. As a result, this matter will be deliberated in the following sections.

4.2 System Model

Because the attention mechanism in FL makes it possible for the parameters of the global model to have the smallest distance possible between themselves and the parameters of the local models, this way, the global model depicts an excellent presentation of all of the local models. In addition, as can be seen in the algorithm 2 for each layer in the global model, the objective is to have the parameters of the global model be as comparable as possible to the local models. This will force FedAtt to consider each client's characteristics.

The algorithm described in 2 is the one that is utilized when applying FedAtt to FL. The Short Seq2Point disaggregation model is being used in this chapter, which was experimented on FedAvg in chapter 3. Here is a recap to the definition of each parameter in 2:

- θ_t: The parameters of global model at time t.
- θ_{t+1}^k: The parameters of k-th client at time $t+1$.
- α_k: The attention-based weights of k-th client.
- w_k^l: The parameters of l-th layer in k-th client.
- w^l: The parameters of global model in l-th layer.

To obtain results that can be compared to FedAvg, FedAtt is applied to both the UK-DALE and REFIT datasets. These datasets have the same data separation described in chapter 3 and they are presented in the tables 4.2, 4.1.

Appliance	Train	Test
Fridge	1, 2, 4	5
Dishwasher	1, 2	5
Washing Machine	1, 5	2
Microwave	1, 2	5

Table 4.1: Data seperation in Federated Attention-based framework with Short Sequence-to-Point model in UK-DALE dataset

Appliance	Train	Test	Validation
Fridge Freezer	2, 5, 6, 9, 12, 13, 15, 19	16	21
Dishwasher	1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 15, 16, 18	20	21
Washing Machine	2, 3, 5, 6, 7, 10, 11, 13, 15, 16, 17	19	20
Microwave	2, 3, 5, 6, 8, 9, 10, 11, 12, 15, 17, 18, 19	4	20

Table 4.2: Data seperation in Federated Attention-based framework with Short Sequence-to-Point model in REFIT dataset

Algorithm 2: FedAtt algorithm

Initialize θ_0 `// Randomly initialized the parameters of global model`

while *global_epoch* > 0 **do**

  $\theta_t \leftarrow$ (Weights of global model in this epoch) ;

  for $k \in$ *client_num* **do**

    Create local_model k;

    $\theta_t^k \leftarrow \theta_t$;

    $\theta_{t+1}^k \leftarrow$ (Train local_model with θ_t^k for chosen number of epochs);

    for $l \in$ *model_layers* **do**

      $\alpha_k^l = softmax(\| w^l - w_k^l \|_p)$;

      `// Calculating the attention-based weight of the` k`-th client in` l`-th layer`

    end

  end

  $\arg\min_{\theta_{t+1}} \sum_{k=1}^{m} [\frac{1}{2} \alpha_k L(\theta_t, \theta_{t+1}^k)^2]$ `// Objective function in FedAtt`

  $\theta_{t+1} \leftarrow \theta_t - \epsilon \sum_{k=1}^{m} \alpha_k (\theta_t - \theta_{t+1}^k)$ `// FedAtt on local parameters`

  Evaluate the global model using θ_{t+1}

end

4.2.1 Differential Privacy

When a model is trained in the standard way, the information about the training data is carried by the model's parameters. To solve this problem, the idea of Differential Privacy (DP) was presented for learning algorithms. The goal is to prevent a trained model from giving away information about whether or not a particular data point was utilized throughout the training process. Although FL adds a layer of privacy to the system, the users' data are still vulnerable to leakage and inverse attack engineering the within the central server. To protect the data from this attack, a Differential Privacy noise with the mean of 0 and the standard deviation of σ can be added to equation 2.13 in FedAvg:

$$w_{t+1} \longleftarrow \sum_{k=1}^{K} \frac{n_k}{N}(w_{t+1}^k + N(0, \sigma^2)) \tag{4.1}$$

And the FedAtt equation 2.16 can be changed to this:

$$\theta_{t+1} \longleftarrow \theta_t - \epsilon \sum_{k=1}^{m} \alpha_k(\theta_t - \theta_{t+1}^k + \gamma N(0, \sigma^2)) \tag{4.2}$$

Adding this white noise to the parameters update equation is a randomized mechanism [17]. The FedAvg equation 4.1 can be modified by including the γ variable from the FedAtt equation 4.2. The magnitude coefficient $\gamma \in (0, 1]$ regulates the impact of randomization on normal noise.

4.2.2 Hyper-parameters Tuning

Most of the recent papers in the area of FL and NILM did not explore hyper-parameter tuning to find the best parameters for the network since FL has lots of parameters and its structure is complex to tune the parameters.

In order to tune the hyper-parameters of FedAtt framework, different parameters were tested with the help of GPU resources available on the Graham and Cedar servers at Compute Canada and the WandB website [64]. The washing machine was used as the experimental appliance. It took approximately four days and 239 possible combinations to investigate, as shown in 4.1. The parameters chosen for tuning and their values are shown in table 4.3.

The goal in WandB was to maximize the F1 score. Based on that, the highest F1 score achieved with the values in table 4.4. The parameters of this table 4.4 then were used in the following experiments.

WandB uses an importance metric to visually display the relationship between the model's hyper-parameters and the metrics by which their performance is measured. The feature importance values for the random forest will be reported after training it with the hyperparameters as inputs and the chosen metric as the target output. Of all the chosen parameters, step size and learning rate got the highest

Batch Size	256, 512, 1024
ϵ	0.1, 0.2, 0.3, 0.4, 0.5, 1
Local Epochs	1, 2
LR	0001, 0.0005, 0.001, 0.005
LR Epsilon	$1e-7$, $1e-8$
Norm p	Frobenius norm, 2-norm

Table 4.3: The chosen values for hyper-parameter tuning (LR: Initial learning Rate, 2-norm: Largest singular value)

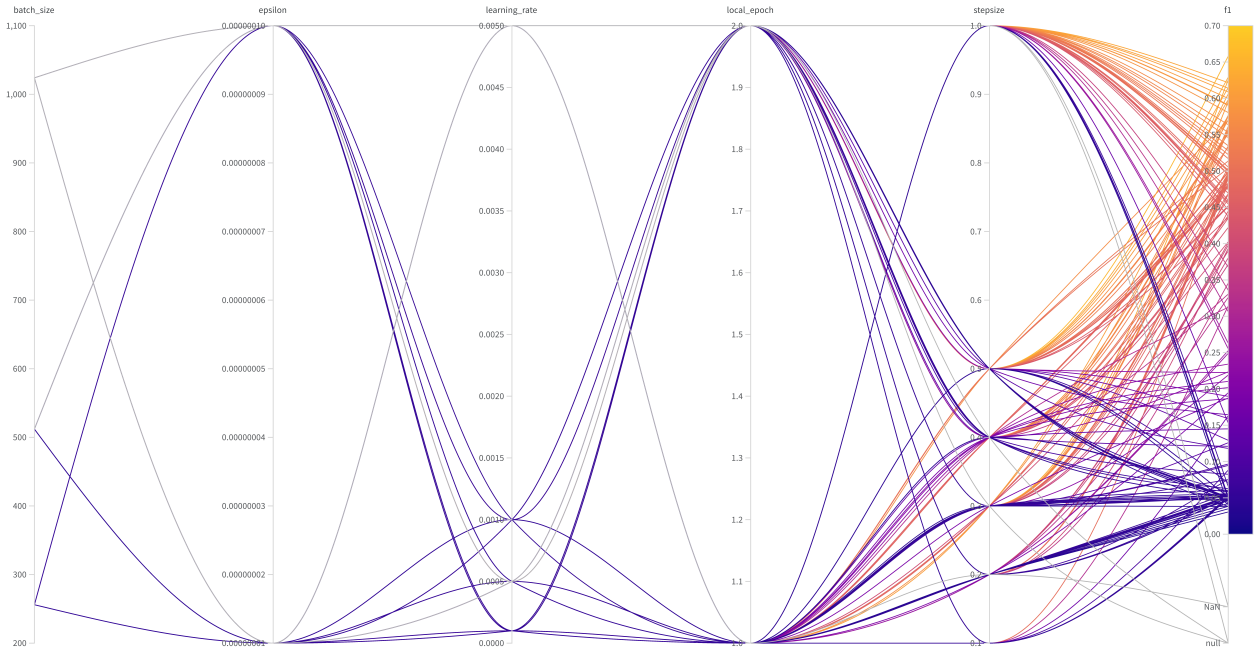

Figure 4.1: 239 combination of parameters for tuning monitored in WandB

Batch Size	LR Epsilon	LR	Step Size	Norm
512	1e-8	0.0005	0.5	Frobenius norm

Table 4.4: The final parameters resulting from hyper-parameter tuning (LR: Learning Rate)

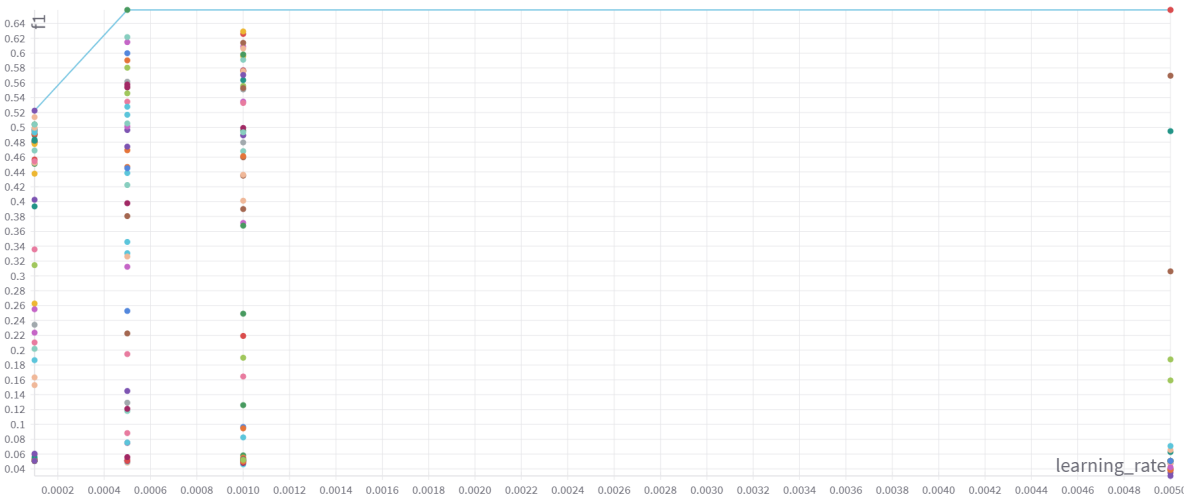

Figure 4.2: Different values of F1 score versus learning rate in WandB

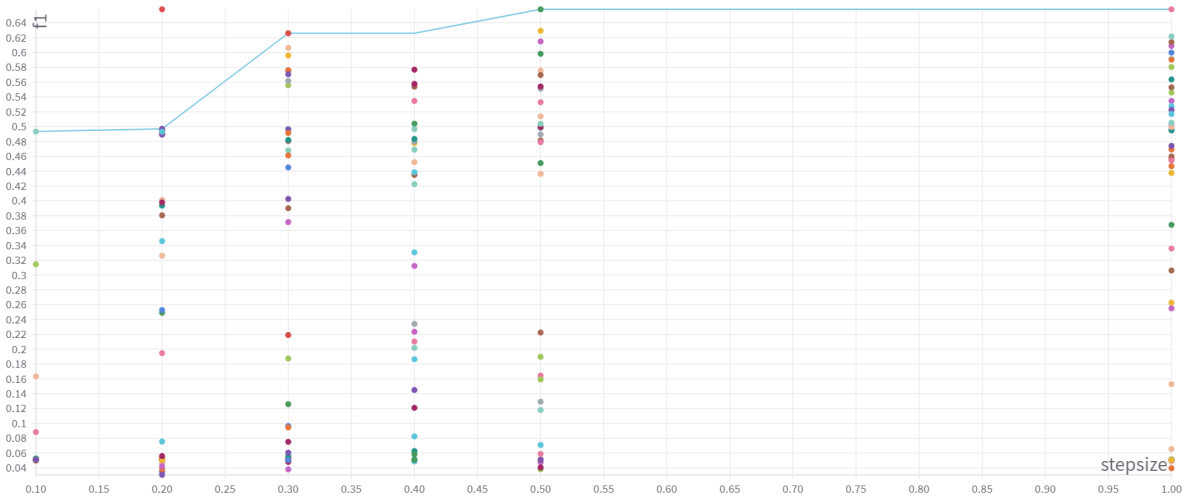

Figure 4.3: Different values of F1 score versus step size in WandB

importance value (0.346, 0.333 respectively). As shown in 4.3, and 4.2, after a certain value for both step size and learning, the value of f1 does not change drastically. Also, for learning rate 4.2, after this certain point, the number of experiments that achieve good performance in f1 decreases.

4.3 Results Analysis

According to figure 4.5, which compares the aggregation methods and centralized model on UK-DALE, the results of FedAtt are mainly comparable with the centralized model. In Microwave, most metrics except RETE and recall are improved. The fridge is the only Appliance in FedAtt with higher results than FedAvg. Having three clients compared to two in other appliances may cause this improvement in attention-based aggregation. However, in the other three appliances, the FedAtt results are in the same range as FedAvg. The reason could be that the number of clients may not be enough to evaluate this framework correctly. Also, FedAtt might need a larger dataset to get enough information on the similarity between the parameters of the local models and global parameters.

	FedAtt				**FedAvg**				**Centr. Short Seq2Point** [32]			
	F	DW	WM	M	F	DW	WM	M	F	DW	WM	M
Acc	**0.65**	**0.97**	0.95	0.97	0.61	0.97	0.92	**0.98**	0.54	0.96	**0.97**	0.91
Pr	**0.51**	0.6	0.18	0.03	0.47	**0.64**	0.13	**0.06**	0.42	0.47	**0.26**	0.01
Re	0.24	**0.53**	0.74	0.42	**0.8**	0.53	**0.75**	0.42	0.74	0.43	0.55	**0.79**
F1	0.33	0.56	0.29	0.05	**0.59**	**0.58**	0.22	**0.1**	0.53	0.45	**0.35**	0.03
MAE	**51**	21	26	**61**	54	**20**	29	61	51	21	**17**	103
RETE	**0.13**	0.1	0.43	0.7	0.29	**0.01**	0.56	0.63	0.29	0.07	**0.28**	**0.16**

Table 4.5: The comparison between Federated Attention-based, Federated Averaging, and centralized Short Sequence-to-Point [32] model with UK-DALE dataset (F: Fridge, Dw: DishWasher, WM: Washing Machine, M: Microwave, Centr.: Centralized)

The results of FedAtt on REFIT are compared to those of FedAvg and centralized Short Seq2Point in the table that can be found at 4.6. The fact that the F1 score in each appliance is higher than the centralized one indicates that the attention mechanism is operating as it should. Despite this, the outcomes of FedAtt and FedAvg fall within the same range. During the experiments conducted in FedAtt, it was noticed that in the first few epochs of the global model, the validation results increased faster compared to FedAvg. However, after that, the changes in the metrics were so small. As a result,

	FedAtt				**FedAvg**				**Centralized Short Seq2Point**			
	FF	DW	WM	M	FF	DW	WM	M	FF	DW	WM	M
Acc	0.54	0.96	0.97	0.99	0.56	**0.98**	**0.98**	0.99	**0.57**	0.94	0.97	0.99
Pr	0.47	0.11	0.47	0.51	0.48	**0.29**	**0.53**	**0.76**	**0.5**	0.1	0.45	0.59
Re	**0.8**	0.29	**0.95**	**0.33**	0.72	0.31	0.87	0.19	0.54	**0.49**	0.93	0.03
F1	**0.59**	0.16	0.63	**0.4**	0.58	**0.3**	**0.66**	0.3	0.25	0.18	0.6	0.06
MAE	47.7	**14.5**	17.2	12.5	45.66	22.2	17.65	**11.49**	**45.86**	22.95	**8.39**	12.37
RETE	0.13	0.64	0.54	**0.46**	**0.04**	**0.21**	0.54	0.64	0.06	0.54	**0.23**	0.7

Table 4.6: The comparison between Federated Attention-based, Federated Averaging, and centralized Short Sequence-to-Point model with REFIT dataset (FF: Fridge Freezer, Dw: DishWasher, WM: Washing Machine, M: Microwave)

this framework might need to be executed for a higher number of global epochs. Unfortunately, because the FedAtt procedure required a significant amount of time to complete, each global epoch with this aggregation method took longer. By increasing the number of global epochs for FedAtt, it will be possible to have sufficient information on each client, resulting in improved performance compared to FedAvg.

This figure 4.4 compares all of the metrics of FedAvg and FedAtt with different fractions of the REFIT dataset. Since in the best case the clients of REFIT dataset is around 13, 50% and 75% of it will be 6 and 9 clients respectively. While the results for FedAvg show a general trend toward improvement, this is not the case for FedAtt. There are several potential explanations for this. One of these is that the window size for this model is short of making the model run faster. On the other hand, this can be a disadvantage for FedAtt because it allows them to reduce the distance between the global and local model weights. The fact that the bulk of the samples in REFIT is 0 indicates that the appliance is turned off. A smaller window size will produce more instances, each of which will have power spent.

The findings depicted in figure 4.5 are highly encouraging. This experiment on UK-DALE using the FedAtt aggregation approach demonstrates that, despite adding more randomization to the local parameters as DP noise, FedAtt is still able to perform in a manner that is comparable to its previous results. In fact, not only is it capable of performing well, but in some appliances (such as the fridge, the microwave, and the dishwasher), it performs even better.

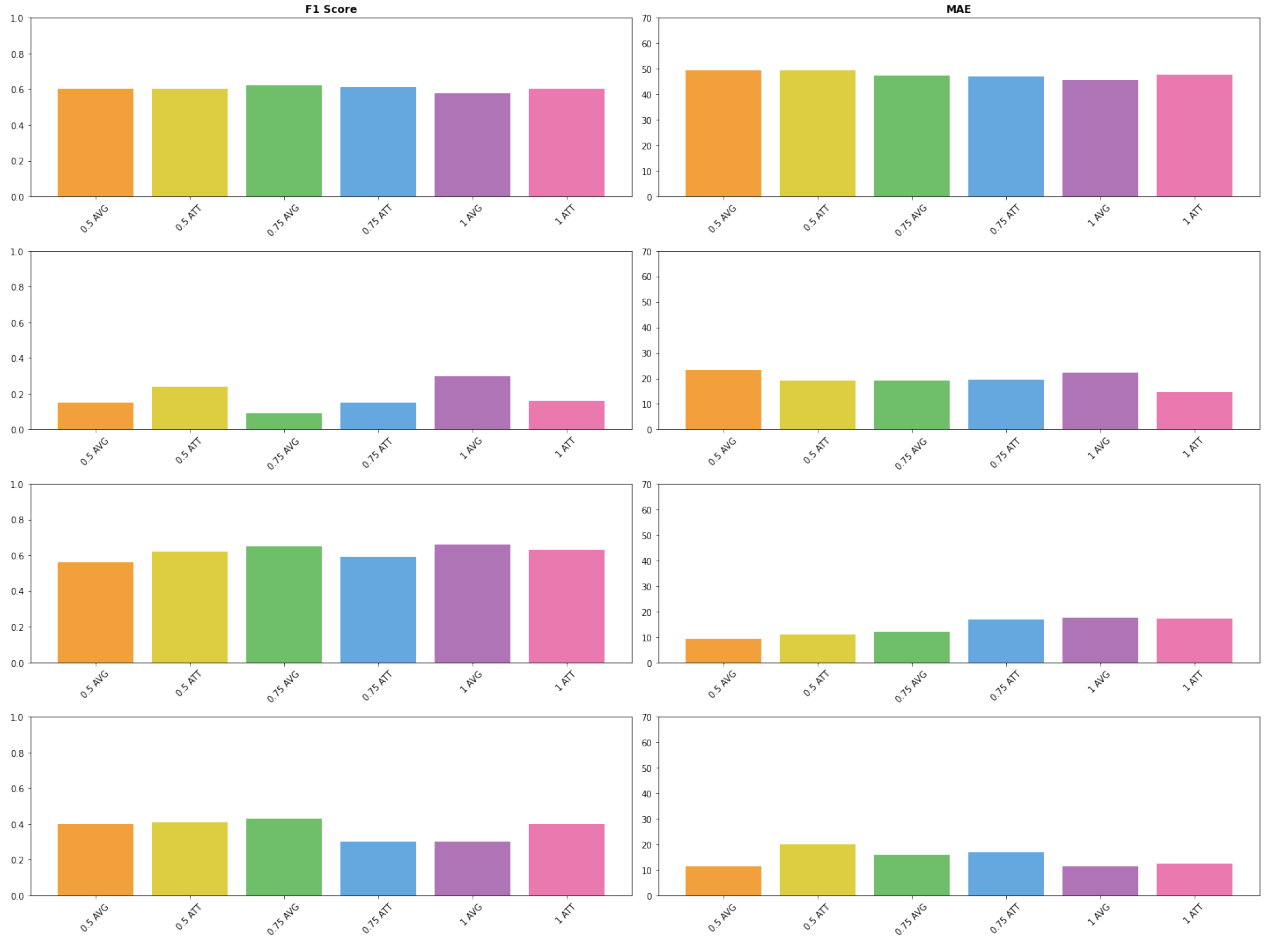

Figure 4.4: Comparing different fractions of clients in REFIT dataset using Federated Averaging and Federated Attention-based methods with Short Sequence-to-Point model

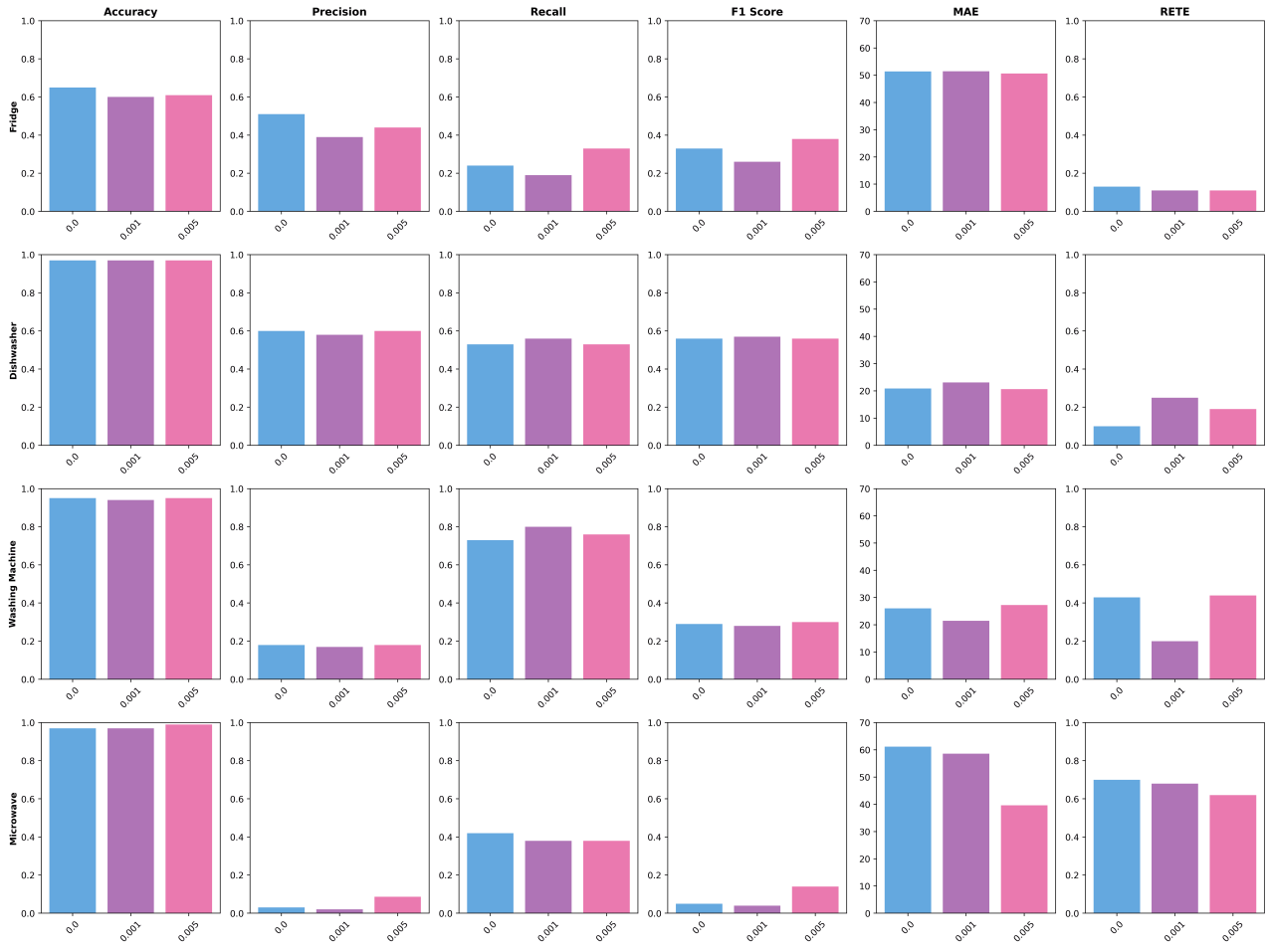

Figure 4.5: Comparing the results of choosing different magnitude coefficient γ for added Differential Privacy noise in UK-DALE dataset using Federated Attention-based method with Short Sequence-to-Point model

Chapter 5

Utilizing Variational Auto-Encoder Model and Federated Approaches for Non-Intrusive Load Monitoring

5.1 Introduction

This chapter provides insight into a recently proposed method for the disaggregation of energy that was only recently published in [33, 58]. The performance of this model is significantly higher than the other proposed methods in this area, such as Seq2Seq, GRU, and Seq2Point, according to the results of [33]. This is a new model in NILM even though this method is a popular approach in machine learning and has been effectively used in NLP and image classification. Previous approaches in NILM have difficulty disaggregating multi-state appliances and generalizing to new households. The performance of NILM techniques for multi-states appliances is poor compared to the other appliances in [65, 30, 27], despite the fact that they may contribute significantly to the energy consumption of the entire house. The disaggregation procedure can yield a generalized network with the assistance of Variational Auto-Encoder [29], since it provides a latent space. The probabilistic encoder makes this approach viable for encoding the information required to reconstruct the consumption of the targeted household appliances. When applied to multi-state appliances, VAE consistently generates more detailed load profiles, which improves the reconstructed power signal.

The VAE is more stable during the training process because it is a generative model. Since its latent space is regularized, it is possible to do interpolations between two distributions learned during

training to get realistic appliance load profiles after decoding. Activations of the same appliance can commonly result in significantly different load profiles. Thus, VAE is well-suited to disaggregating energy to generate the target appliance's power signal.

As was covered in the previous chapters, this thesis aims to combine the FL framework with NILM in a way that not only improves the privacy of the clients and reduces the data transmission cost but also performs comparably with the centralized model. Short Seq2Point was first chosen as a disaggregation approach in FL with small appliance window sizes [25]. However, the lack of generalizability was a reason for searching for a better model with higher performance. In the following sections, the details of this combination and its performance will be elaborated.

5.2 System Model

As it was explained in chapter 2, this model contains two major parts (encoder and decoder), which are shown in figure 5.1. $x = x_1, x_2, ..., x_T$ here is the vector of main power or the aggregated power samples of the whole household, and $y = y_1, y_2, ..., y_T$ is the vector corresponding values in that same time steps for the targeted appliance. T is the length of the window size for the appliance. The model 5.1 first tries to convert the aggregated power x to latent space z, and from that point, it reconstructs the output, which is the targeted appliance's signal. It should be noted that unlike Short Seq2Point [25], where the model's output is the midpoint element of the target appliance's corresponding window, in VAE, y has the same number of samples as x.

The IBN-Net [45] layer in 5.1 contains three successive convolution layers combined with batch normalization and a Rectified Linear Unit (ReLU) activation function. The normalization layer in IBN-Net can help to gain generalizability for the model. A residual link connects the IBN-Net input to the instance normalization layer to avoid the problem of vanishing gradients and improve the flow of gradients throughout model training. The skip connections from IBN-Net in the encoder to the decoder same as the U-Net architecture [52], can improve the learning ability of the model since the decoder now has some extra information on the feature maps in the encoder. With skip connections, details from the combined power can be used to improve the ability to reconstruct the signal from the target appliance.

VAE model will be used in the FL framework with two available aggregation methods; FedAvg and FedAtt. To compare the results with the original paper [33], this thesis follows the same data separation for training and testing. The appliances in this experiment are fridge, dishwasher, washing machine, microwave, and kettle. For all the appliances, 80% of house 2 data is used for testing and 20%

for validation. Houses 1 and 5 are used as clients for FL. The window size (T) for all the appliances is 1024. This large window size, unlike the choice of window sizes in chapter 4 and 3 gives the model a better understanding of how long the appliance was active. The optimizer chosen for VAE is Root Mean Square Propagation (RMSP). The initialized value of the learning rate in local models is 0.001, decreasing based on the global epoch. Batch size and stride for multi-state appliances are 256 and 32 and for the other appliances are 64 and 150 based on [33]. The global epoch is set to 50, and the local epoch for clients is 2. The step size for FedAtt is 0.5, and p norm is Frobenius.

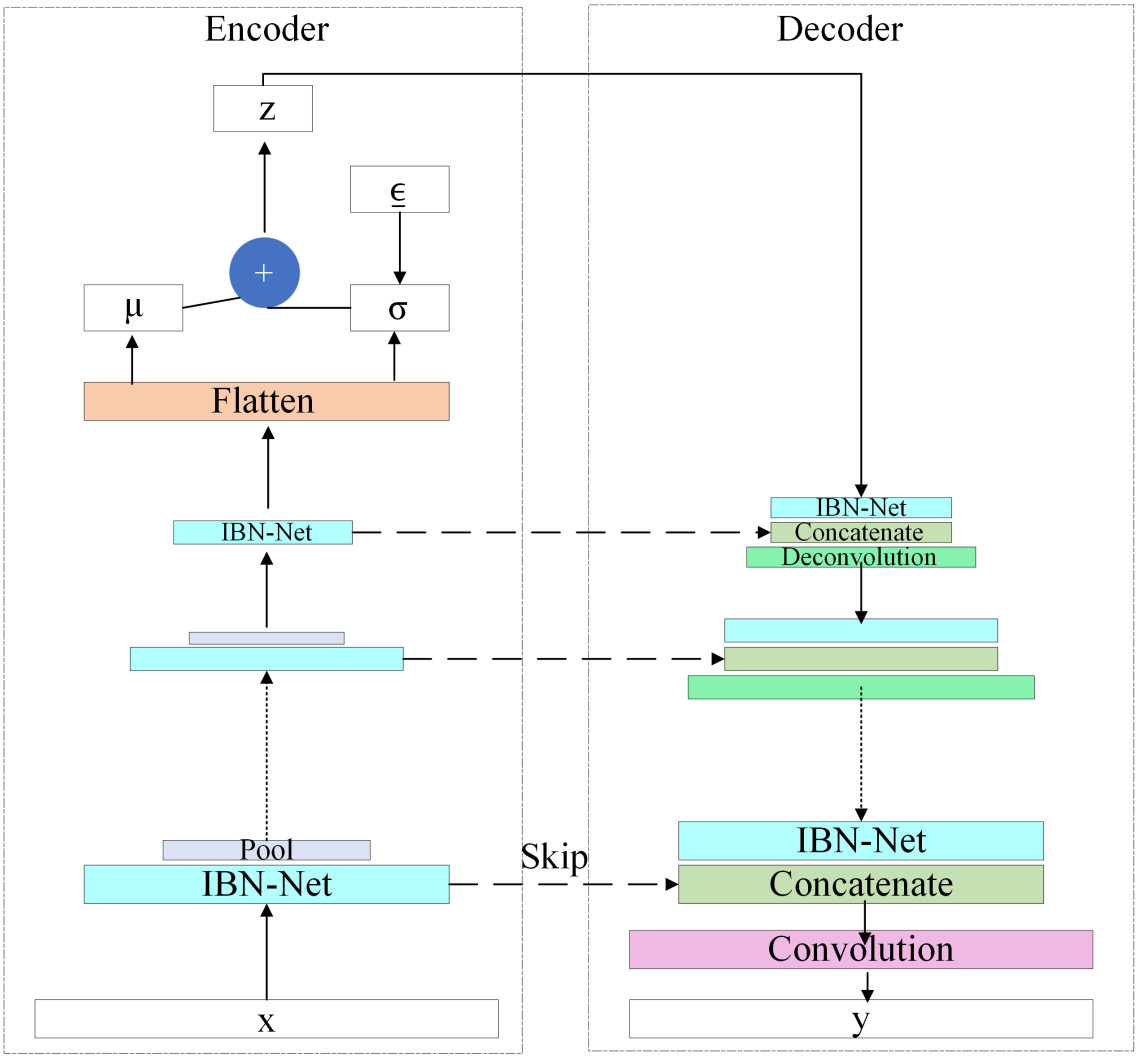

Figure 5.1: Variational Auto-Encoder model

5.3 Results Analysis

The results of FedAtt and FedAvg are compared with [33] in the table 5.1. The results that are written in italics in FedAtt are greater than FedAvg. The values that are shown in bold and italic are the ones that are greater than those found in the original paper and in the FedAvg. Both of FL's techniques for aggregation, as indicated in the table, produce results that are comparable to those of centralized VAE when it comes to all of the metrics. The fact that the numbers are so close demonstrates that a generalized model such as VAE can even assist FL in achieving better results and performance. FedAtt produces better outcomes than the centralized model in both multi-state appliances (washing machine, dishwasher and microwave), which demonstrates how important it is to pay attention to the parameters that are unique to each individual client. Aside from that, this advancement demonstrates that having appropriate weights can contribute to the framework performing well. Another significant improvement can be seen in the microwave. The attention-based aggregation that is being used is assisting the framework in producing results that are superior to the FedAvg.

The table 5.2 was made to give a more detailed illustration of the benefits that may be gained from utilizing various approaches and models in FL. The performance of FL with VAE is superior across all the appliances and metrics. As a result, this demonstrates that a generalized model like VAE can produce considerably better results. On the other hand, Short Seq2Point works to simplify the network in order to lower the number of input window samples and model parameters, all while attempting to deliver the result in a shorter amount of time. In Short Seq2Point, each appliance had its window size related to its power consumption characteristics and state changes. Still, in VAE, all appliances have the same window size. The VAE model is a complex one that contains many skip connections and parameters. Consequently, there is always a trade-off between the expense of parameter transfer between clients and servers and the cost of losing the ability to do VAE calculations in a real-time manner.

Since the F1 score incorporates both precision and recall within itself, it is a better metric to show the improvements in the results when identifying the state of the appliances. Also, MAE shows how accurate are the predictions of energy consumption for each appliance. Thus, to show the average changes in the metrics compared to the centralized counterpart, these two metrics are used. for F1 score and MAE, the changes in the results in FedAtt VAE, FedAvg VAE, FedAtt Short Seq2Point and FedAvg Seq2Point are: -13%, -12%, +7% and +58% for F1 and -35%, -36%, -2%, -7% in MAE. Thus, although the overall results in VAE are better than Short Seq2Point, FL with Short Seq2Point provided

		Pr	Re	F1	MAE
F	FedAtt	0.9	*0.83*	*0.86*	*16.1*
	FedAvg	0.9	0.8	0.85	17.5
	VAE [33]	0.91	0.88	0.89	15.1
DW	FedAtt	0.98	***0.81***	***0.89***	18.1
	FedAvg	0.98	0.8	0.88	14.6
	VAE [33]	0.95	0.76	0.84	11.6
WM	FedAtt	0.67	***0.97***	0.8	7.8
	FedAvg	0.85	0.95	0.9	6.6
	VAE [33]	0.86	0.95	0.9	6.2
M	FedAtt	0.79	*0.26*	*0.4*	*7.9*
	FedAvg	0.85	0.18	0.3	10.2
	VAE [33]	0.84	0.46	0.59	5.1
K	FedAtt	0.94	***0.99***	0.96	8.9
	FedAvg	0.96	0.99	0.98	7.78
	VAE [33]	0.97	0.98	0.97	6.1

Table 5.1: Comparing the results of Federated Attention-based and Federated Averaging with Variational Auto-Encoder model with original paper [33] on UK-DALE dataset (F: Fridge, K: Kettle, M: Microwave, Dishwasher: DW, WM: Washing Machine, Pr: Precision, Re: Recall)

the best enhancement in the results compared to the centralized versions.

		Pr	Re	F1	MAE
F	FedAtt VAE	0.9	0.83	0.86	16.1
	FedAtt S S2P	0.51	0.24	0.33	51
	FedAvg VAE	0.9	0.8	0.85	17.5
	FedAvg S S2P	0.47	0.8	0.59	54
	Centralized VAE [33]	**0.91**	**0.88**	**0.89**	**15.1**
	Centralized S S2P [32]	0.42	0.74	0.53	51
DW	FedAtt VAE	**0.98**	**0.81**	**0.89**	18.1
	FedAtt S S2P	0.6	0.53	0.56	21
	FedAvg VAE	0.98	0.8	0.88	14.6
	FedAvg S S2P	0.64	0.53	0.58	20
	Centralized VAE [33]	0.95	0.76	0.84	**11.6**
	Centralized S S2P [32]	0.47	0.43	0.45	21
WM	FedAtt VAE	0.67	**0.97**	0.8	7.8
	FedAtt S S2P	0.18	0.74	0.29	26
	FedAvg VAE	0.85	0.95	0.9	6.6
	FedAvg S S2P	0.13	0.75	0.22	29
	Centralized VAE [33]	**0.86**	0.95	**0.9**	**6.2**
	Centralized S S2P [32]	0.26	0.55	0.35	17
M	FedAtt VAE	0.79	0.26	0.4	7.9
	FedAtt S S2P	0.03	0.42	0.05	61
	FedAvg VAE	**0.85**	0.18	0.3	10.2
	FedAvg S S2P	0.06	0.42	0.1	61
	Centralized VAE [33]	0.84	0.46	**0.59**	**5.1**
	Centralized S S2P [32]	0.01	**0.79**	0.03	103

Table 5.2: Comparing all the results from chapter 3, 4 and 5 with UK-DALE dataset (F: Fridge, K: Kettle, M: Microwave, Dishwasher: DW, WM: Washing Machine, Pr: Precision, Re: Recall, S S2P: Short Seq2Point)

Figures 5.2 and 4.5 show the impact of using different values for γ for added noise in the federated optimization. This noise keeps parameters safe while they are in the central server. With increasing the value of γ, the results of FedAvg 5.2 slightly fluctuate; however, FedAtt keeps the results steadier. The results surprisingly increase with greater γ value in some cases, such as fridge in FedAvg and kettle, microwave, and dishwasher in FedAtt. The reason for that is that adding randomization to the weights can help the model learn better and be ready for different parameter values. However, with having 3 appliances out of 5 with better results in $\gamma = 0.01$, FedAtt can better maintain this noise in the updating process.

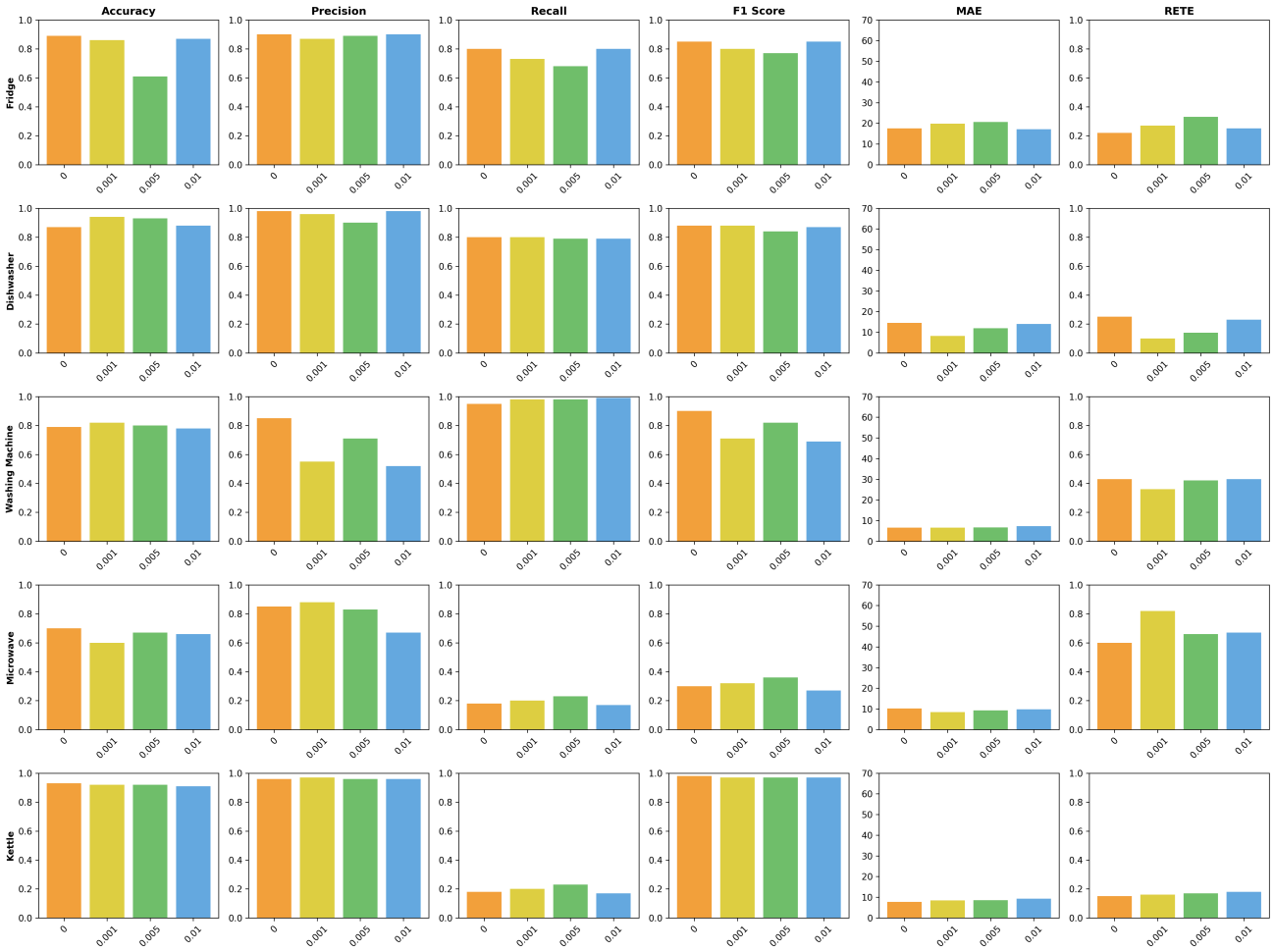

Figure 5.2: Comparing the results of different magnitude coefficient γ for added Differential Privacy noise in UK-DALE dataset using Federated Averaging method with Variational Auto-Encoder model

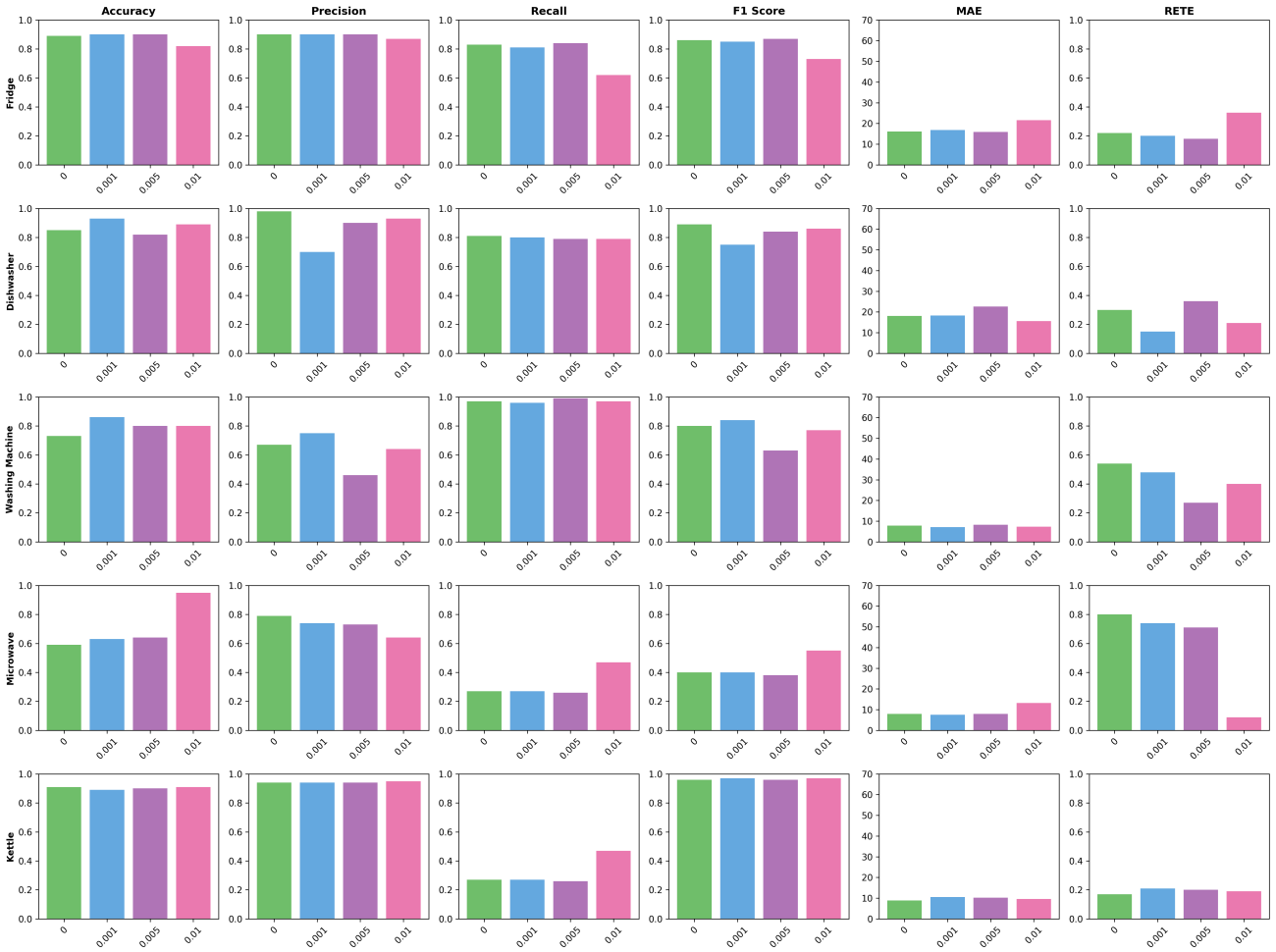

Figure 5.3: Comparing the results of different magnitude coefficient γ for added Differential Privacy noise in UK-DALE dataset using Federated Attention-based method with Variational Auto-Encoder model

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This thesis presented several distributed privacy-preserving strategies for NILM based on FL to ensure users' privacy and reduce communication-cost.

The resulting framework is applicable for real-time energy disaggregation. Also, it significantly decreases data communication from the client's end to the global model due to the combination of FL and the different disaggregation models. The weight is the only parameter shared between the local and global models in all the proposed methods.

According to chapter 3 [25], using FedAvg on Short Seq2Point for energy disaggregation can not only provide a secure and privacy-preserving system that requires less storage capacity in the central server, but it can also perform better than the centralized model in some cases. To better represent the performance of this framework, the REFIT dataset with 20 households has been used to explore the importance of the quantity of clients in the system. Different fractions of clients were chosen for FedAvg in REFIT, and the results proved that this framework can still provide comparable results to the centralized model in each fraction.

In chapter 4, another aggregation method called FedAtt was introduced for energy disaggregation, which considers the characteristics of each client. The global parameters in FedAtt in a way that the weights in the global model have the least distance to local weights and can effectively represent each client's attributes in the central server. In both UK-DALE and REFIT, FedAtt provides comparable results to the centralized model. The findings are very close to FedAvg in certain cases, possibly due to the necessity for higher global epochs or a dataset with more active states for appliances. In this

chapter, hyperparameter tuning is performed, a time-consuming and difficult procedure in FL, with the assistance of WandB and Compute Canada. The resulting parameters produce the best results in FedAtt. Besides, for more privacy in the parameter transmission between clients and servers, the γ parameter was used as the magnitude coefficient for added noise in the federated optimization equation. The results showed that, even with a different randomization value for γ in FedAtt, it could perform well and adapt to the new parameters.

Finally, in chapter 5, the new approach in NILM called VAE was used to generalize the model, especially for the multi-state appliances. With a much wider window size in VAE, both FedAtt and FedAvg outperformed their previous model in all the metrics. FedAtt with VAE outperformed Short Seq2Point because now it has more detailed parameters and information due to the model's complexity. In both FedAvg and FedAtt, with adding Differential Privacy noise to protect the client's privacy, the system still provides excellent results. FedAtt results contain less fluctuation than FedAvg in all the metrics, which shows VAE model helps the Federated Attention-based to maintain the results in the same range and adapt better to different parameter values.

6.2 Future Work

In this section, some interesting and potentially fruitful ideas that could be used to extend and improve the work are presented, as well as some questions regarding possible model improvements that could be required to put a plan of this nature into practice in the real world.

6.2.1 Window Size

In energy disaggregation models, the window size is an essential element. While some appliances have a short activation time, others can have the opposite characteristic. Although in the Seq2Point model, each appliance had its suitable window size, choosing the same duration in window size for REFIT appliances as UK-DALE may not be the most efficient choice. Thus, to keep producing improved outcomes with the VAE and all the other models for each appliance, it is necessary to investigate this component based on the structure of the datasets.

6.2.2 Transfer Learning

Transfer Learning (TL) needs to be used to make the models more flexible for the new residences. The energy consumption for different appliances varies in different countries and areas. TL can help the model be trained more quickly. It can also help the model offer better results because it has already been pretrained with a different dataset and has extracted the valuable features of that chosen appliance for the model. Deploying TL in FL for energy disaggregation has the potential to expedite the development

of a rapid, general method.

6.2.3 Meta Learning

Rapid adaptation of trained models to the new distribution of NILM datasets is possible with the help of meta-learning. As presented in [35] the performance of FedAvg combined with Meta-learning has significantly increased compared to the centralized model. Meta-learning can be an alternative to TL, and to evaluate the performance of FedAtt, these two methods can be combined and provide promising results.

6.2.4 Number of Clients in Federated Learning

When more clients are involved in FL, the system can more accurately replicate how it would function in the real world. Investigating the possibility of combining various datasets to include more households is crucial. Also, various techniques should be utilized for augmenting the data based on the other buildings.

6.2.5 The Choice of Appliances

Although various appliances can be found in a household, only a few of them are considered for inclusion in the NILM. This is because the rest of the appliances make a more negligible contribution to the aggregated total amount of power usage. However, in the future, to have an IoT-based NILM system, other appliances should also be explored in NILM. This should be done to establish a viable system that is flexible enough to handle newly released home devices.

6.2.6 Thresholding in Appliances

For computing the parameters, such as accuracy, F1, precision, and recall, the chosen threshold at which the state of the appliance changes to ON can seriously affect the outcomes of a model. Multi-state appliances, such as dishwasher and washing machine, make this possibility much more important. Therefore, depending on the dataset, picking the suitable threshold for each device could affect the model's overall performance conclusion.

6.2.7 Behavioral Analytics

As discussed in chapter 2, behavioral analytics is important in determining how much energy a given household uses. Every individual has their own routines and preferences regarding using their home appliances. These patterns can even fluctuate according to the geographic situation and construction of their houses, the inhabitants' community, the weather, and the season. Deploying specific models to extract these patterns from data beforehand can improve federated optimization stages and give more accurate parameters to the central server.

6.2.8 Differential Privacy

Differential Privacy was used in both FedAvg and FedAtt as an added noise. This noise is added to each client's parameters while aggregating them on the central server. Some encoding methods can also be deployed to secure the transmission of these parameters to the central server. Moreover, secure multi-layer computation [19], Fed-SMP [22], and OLIVE [26] can be explored for further enhancing the privacy of FL in NILM.

Bibliography

[1] Martin Abadi et al. "Deep learning with differential privacy". In: *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016, pp. 308–318.

[2] Monica Anderson. "Technology device ownership: 2015". In: (2015).

[3] The Irish Social Science Data Archive. *Data from the Commission for Energy Regulation*. `https://www.ucd.ie/issda/data/commissionforenergyregulationcer/`. (Accessed on 03/08/2022). 2012.

[4] Christos L Athanasiadis et al. "Real-Time Non-Intrusive Load Monitoring: A Machine-Learning Approach for Home Appliance Identification". In: *2021 IEEE Madrid PowerTech*. IEEE. 2021, pp. 1–6.

[5] Elnaz Azizi et al. "Residential household non-intrusive load monitoring via smart event-based optimization". In: *IEEE Transactions on Consumer Electronics* 66.3 (2020), pp. 233–241.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[7] Alfonso Capozzoli, Marco Savino Piscitelli, and Silvio Brandi. "Mining typical load profiles in buildings to support energy management in the smart city context". In: *Energy Procedia* 134 (2017), pp. 865–874.

[8] Fei Chen et al. "Federated meta-learning with fast convergence and efficient communication". In: *arXiv preprint arXiv:1802.07876* (2018).

[9] Kunjin Chen, Jun Hu, and Ziyu He. "Data-driven residential customer aggregation based on seasonal behavioral patterns". In: *2017 IEEE Power & Energy Society General Meeting*. IEEE. 2017, pp. 1–5.

[10] Junyoung Chung et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: *arXiv preprint arXiv:1412.3555* (2014).

[11] Michele D'Incecco, Stefano Squartini, and Mingjun Zhong. "Transfer learning for non-intrusive load monitoring". In: *IEEE Transactions on Smart Grid* 11.2 (2019), pp. 1419–1429.

[12] Shuang Dai et al. "FederatedNILM: A Distributed and Privacy-preserving Framework for Non-intrusive Load Monitoring based on Federated Deep Learning". In: *arXiv preprint arXiv:2108.03591* (2021).

[13] John C Duchi, Michael I Jordan, and Martin J Wainwright. "Privacy aware learning". In: *Journal of the ACM (JACM)* 61.6 (2014), pp. 1–57.

[14] Cynthia Dwork. "Differential privacy: A survey of results". In: *International conference on theory and applications of models of computation*. Springer. 2008, pp. 1–19.

[15] Cynthia Dwork, Aaron Roth, et al. "The algorithmic foundations of differential privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (2014), pp. 211–407.

[16] Sannara Ek et al. "Evaluating Federated Learning for human activity recognition". In: *Workshop AI for Internet of Things, in conjunction with IJCAI-PRICAI 2020*. 2021.

[17] Robin C Geyer, Tassilo Klein, and Moin Nabi. "Differentially private federated learning: A client level perspective". In: *arXiv preprint arXiv:1712.07557* (2017).

[18] Antonious Girgis et al. "Shuffled model of differential privacy in federated learning". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 2521–2529.

[19] Slawomir Goryczka, Li Xiong, and Vaidy Sunderam. "Secure multiparty aggregation with differential privacy: A comparative study". In: *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. 2013, pp. 155–163.

[20] Andrew Hard et al. "Federated learning for mobile keyboard prediction". In: *arXiv preprint arXiv:1811.03604* (2018).

[21] George William Hart. "Nonintrusive appliance load monitoring". In: *Proceedings of the IEEE* 80.12 (1992), pp. 1870–1891.

[22] Rui Hu, Yanmin Gong, and Yuanxiong Guo. "Federated learning with sparsified model perturbation: Improving accuracy under client-level differential privacy". In: *arXiv preprint arXiv:2202.07178* (2022).

[23] Shaoxiong Ji et al. “Learning private neural language modeling with attentive aggregation”. In: *2019 International joint conference on neural networks (IJCNN)*. IEEE. 2019, pp. 1–8.

[24] Yihan Jiang et al. “Improving federated learning personalization via model agnostic meta learning”. In: *arXiv preprint arXiv:1909.12488* (2019).

[25] Shamisa Kaspour and Abdulsalam Yassine. “A Federated Learning Model With Short Sequence To Point Mechanism For Smart Home Energy Disaggregation”. In: *2022 IEEE Symposium on Computers and Communications (ISCC)*. 2022, pp. 1–6. DOI: 10.1109/ISCC55528.2022.9912852.

[26] Fumiyuki Kato, Yang Cao, and Masatoshi Yoshikawa. “OLIVE: Oblivious and Differentially Private Federated Learning on Trusted Execution Environment”. In: *arXiv preprint arXiv:2202.07165* (2022).

[27] Jack Kelly and William Knottenbelt. “Neural nilm: Deep neural networks applied to energy disaggregation”. In: *Proceedings of the 2nd ACM international conference on embedded systems for energy-efficient built environments*. 2015, pp. 55–64.

[28] Jack Kelly and William Knottenbelt. “The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes”. In: *Scientific Data* 2.150007 (2015). DOI: 10.1038/sdata.2015.7.

[29] Diederik P Kingma and Max Welling. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

[30] Weicong Kong et al. “A practical solution for non-intrusive type II load monitoring based on deep learning and post-processing”. In: *IEEE Transactions on Smart Grid* 11.1 (2019), pp. 148–160.

[31] Alex Krizhevsky, Geoffrey Hinton, et al. “Learning multiple layers of features from tiny images”. In: (2009).

[32] Odysseas Krystalakos, Christoforos Nalmpantis, and Dimitris Vrakas. “Sliding window approach for online energy disaggregation using artificial neural networks”. In: *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*. 2018, pp. 1–6.

[33] Antoine Langevin et al. “Energy disaggregation using variational autoencoders”. In: *Energy and Buildings* 254 (2022), p. 111623.

[34] Qi Li et al. “Energy Disaggregation with Federated and Transfer Learning”. In: *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*. IEEE. 2021, pp. 698–703.

[35] Ruohong Liu and Yize Chen. "Learning Task-Aware Energy Disaggregation: a Federated Approach". In: *arXiv preprint arXiv:2204.06767* (2022).

[36] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. "Effective approaches to attention-based neural machine translation". In: *arXiv preprint arXiv:1508.04025* (2015).

[37] Stephen Makonin et al. "Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014". In: *Scientific data* 3.1 (2016), pp. 1–12.

[38] Brendan McMahan et al. "Communication-Efficient Learning of Deep Networks from Decentralized Data". In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. Ed. by Aarti Singh and Jerry Zhu. Vol. 54. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1273–1282. URL: https://proceedings.mlr.press/v54/mcmahan17a.html.

[39] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[40] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. "Recurrent models of visual attention". In: *Advances in neural information processing systems* 27 (2014).

[41] David Murray, Lina Stankovic, and Vladimir Stankovic. "An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study". In: *Scientific data* 4.1 (2017), pp. 1–12.

[42] Christoforos Nalmpantis and Dimitris Vrakas. "On time series representations for multi-label NILM". In: *Neural Computing and Applications* 32.23 (2020), pp. 17275–17290.

[43] Christoforos Nalmpantis and Dimitris Vrakas. "Signal2vec: Time series embedding representation". In: *International conference on engineering applications of neural networks*. Springer. 2019, pp. 80–90.

[44] Bishnu Nepal et al. "Electricity load forecasting using clustering and ARIMA model for energy management in buildings". In: *Japan Architectural Review* 3.1 (2020), pp. 62–76.

[45] Xingang Pan et al. "Two at once: Enhancing learning and generalization capacities via ibn-net". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 464–479.

[46] Oliver Parson et al. "Dataport and NILMTK: A building data set designed for non-intrusive load monitoring". In: *2015 ieee global conference on signal and information processing (globalsip)*. IEEE. 2015, pp. 210–214.

[47] Matthias Paulik et al. "Federated evaluation and tuning for on-device personalization: System design & applications". In: *arXiv preprint arXiv:2102.08503* (2021).

[48] Veronica Piccialli and Antonio M Sudoso. "Improving non-intrusive load disaggregation through an attention-based deep neural network". In: *Energies* 14.4 (2021), p. 847.

[49] Jacob Poushter et al. "Smartphone ownership and internet usage continues to climb in emerging economies". In: *Pew research center* 22.1 (2016), pp. 1–44.

[50] Swaroop Ramaswamy et al. "Federated learning for emoji prediction in a mobile keyboard". In: *arXiv preprint arXiv:1906.04329* (2019).

[51] Attique Ur Rehman et al. "Comparative Evaluation of Machine Learning Models and Input Feature Space for Non-intrusive Load Monitoring". In: *Journal of Modern Power Systems and Clean Energy* 9.5 (2021), pp. 1161–1171.

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[53] Cynthia Rosenzweig et al. "Attributing physical and biological impacts to anthropogenic climate change". In: *Nature* 453.7193 (2008), pp. 353–357.

[54] Singh Shailendra and Yassine Abdulsalam. "IoT Big Data Analytics with Fog Computing for Household Energy Management in Smart Grids". In: *Smart Grid and Internet of Things Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Vol. 256. Springer, 2019.

[55] Shailendra Singh and Abdulsalam Yassine. "Big data mining of energy time series for behavioral analytics and energy consumption forecasting". In: *Energies* 11.2 (2018), p. 452.

[56] Shailendra Singh and Abdulsalam Yassine. "Mining Energy Consumption Behavior Patterns for Households in Smart Grid". In: *IEEE Transactions on Emerging Topics in Computing* 7.3 (2019), pp. 404–419. DOI: 10.1109/TETC.2017.2692098.

[57] Shailendra Singh, Abdulsalam Yassine, and Rachid Benlamri. "Towards Hybrid Energy Consumption Prediction in Smart Grids with Machine Learning". In: *2018 4th International Conference on Big Data Innovations and Applications (Innovate-Data)*. 2018, pp. 44–50. DOI: 10.1109/Innovate-Data.2018.00014.

[58] Tharmakulasingam Sirojan, B Toan Phung, and Eliathamby Ambikairajah. "Deep neural network based energy disaggregation". In: *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*. IEEE. 2018, pp. 73–77.

[59] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems*. 2014, pp. 3104–3112.

[60] Andrew Tittaferrante and Abdulsalam Yassine. "Importance Scaling for Elastic Appliance for Automated Power Management in Smart Homes". In: *2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT IoT and AI (HONET-ICT)*. 2019, pp. 115–120. DOI: 10.1109/HONET.2019.8907970.

[61] Jacopo Torriti. "Temporal aggregation: time use methodologies applied to residential electricity demand". In: *Utilities Policy* 64 (2020), p. 101039.

[62] Nikolaos Virtsionis-Gkalinikis, Christoforos Nalmpantis, and Dimitris Vrakas. "SAED: Self-attentive energy disaggregation". In: *Machine Learning* (2021), pp. 1–20.

[63] Haijin Wang, Caomingzhe Si, and Junhua Zhao. "A Federated Learning Framework for Non-Intrusive Load Monitoring". In: *arXiv preprint arXiv:2104.01618* (2021).

[64] *Weights and Biases The developer-first MLOps platform*. https://wandb.ai/site. Accessed: 2022-09-10.

[65] Qian Wu and Fei Wang. "Concatenate convolutional neural networks for non-intrusive load monitoring across complex background". In: *Energies* 12.8 (2019), p. 1572.

[66] Ting Yang, Minglun Ren, and Kaile Zhou. "Identifying household electricity consumption patterns: A case study of Kunshan, China". In: *Renewable and Sustainable Energy Reviews* 91 (2018), pp. 861–868.

[67] Wenpeng Yin et al. "Abcnn: Attention-based convolutional neural network for modeling sentence pairs". In: *Transactions of the Association for Computational Linguistics* 4 (2016), pp. 259–272.

[68] Chaoyun Zhang et al. "Sequence-to-point learning with neural networks for non-intrusive load monitoring". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[69] Hengshuang Zhao et al. "Pyramid scene parsing network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2881–2890.

[70] Yuchen Zhao et al. “Semi-supervised Federated Learning for Activity Recognition”. In: *arXiv preprint arXiv:2011.00851* (2020).