Feature-Centric Approaches to Non-Intrusive Load Monitoring and Appliance Identification

Muhammad Asad

A Thesis

in

The Department

of

Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Quality Systems Engineering) at

Concordia University

Montréal, Québec, Canada

August 2025

© Muhammad Asad, 2025

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Muhammad Asad**

Entitled: **Feature-Centric Approaches to Non-Intrusive Load Monitoring and Appliance Identification**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Quality Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair and Examiner
Dr. Abdessamad Ben Hamza

_____ Examiner
Dr. Zachary Patterson

_____ Supervisor
Dr. Nizar Bouguila

_____ Supervisor
Dr. Manar Amayri

Approved by _____
Chun Wang, Chair
Department of Concordia Institute for Information Systems Engineering

_____ 2025

_____
Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Feature-Centric Approaches to Non-Intrusive Load Monitoring and Appliance Identification

Muhammad Asad

Load disaggregation refers to estimating appliance-level consumption from overall household energy data. It includes tasks like load identification and energy disaggregation. Researchers are actively developing various machine learning and deep learning techniques to disaggregate total household energy consumption into appliance-level usage. At the same time, many are focusing on identifying individual appliance loads to detect faulty devices or to improve the overall disaggregation process. This thesis makes two significant contributions to the field, addressing the challenges of total load separation and appliance identification. The first contribution focuses on energy disaggregation using a simplified Feed-Forward Neural Network architecture optimized for performance and efficiency. Oversampling techniques are developed for training data to improve the detection of appliance activation cycles. Furthermore, the model incorporates additional features derived from aggregate consumption profiles, enhancing input diversity and robustness. This approach is tested on the RAE, REFIT, and REDD datasets under both clean and noisy conditions. The second contribution addresses appliance-level load identification using a Kolmogorov–Arnold Network, offering a lightweight and efficient alternative to deep models. Around 75 features are extracted from voltage and current signals, grouped into statistical, power-related, and frequency-domain categories. An effective feature selection process is conducted using multiple tests and correlation matrices to retain only the most informative inputs, thereby reducing model complexity and enhancing generalization. Additionally, we tune the hyperparameters of the KAN to control the degree of oversampling, allowing it to better handle imbalanced data. The model is evaluated using three public datasets: COOLL, PLAID, and WHITED.

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. Nizar Bouguila, for his support, guidance, and encouragement throughout my research at Concordia University. His professional expertise and contributions have been instrumental in the advancement and overall quality of this thesis.

I am truly thankful to my supervisor, Dr. Manar Amayri, for her support, helpful advice, and steady encouragement during my research at Concordia University. Her knowledge and consistent efforts greatly improved the development and quality of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Energy consumption is a fundamental driver of modern life; however, it also produces externalities that must be acknowledged and addressed. About 73% of global electricity comes from fossil fuels and nuclear power, with coal making up 36.4% Akbar et al. (2024). Nearly 60% is consumed by homes and businesses Akbar et al. (2024); Faustine, Mvungi, Kaijage, and Kisangiri (2017), leading to environmental concerns like CO2 emissions and global warming Kelly and Knottenbelt (2015). Recent studies show that providing appliance-level energy data to consumers can reduce annual consumption by up to 12% Akbar et al. (2024); Bonfigli et al. (2018). Non-Intrusive Load Monitoring (NILM) is the process of obtaining the energy usage of each appliance from a single metering site Akbar et al. (2024); Bucci, Ciancetta, Fiorucci, and Mari (2020). The potential for energy savings arises from a combination of factors involving both residential consumers and energy providers. Due to increasing usage of electricity, limited energy resources, and increasing demand for electrical appliances, it has become essential for users to monitor their energy usage and manage it. For consumers, having detailed data on appliance energy usage empowers them to take steps to lower their bills, such as by replacing older, inefficient devices with more energy-efficient models. Meanwhile, energy providers can use this data to forecast energy demand more accurately, implement improved management strategies, and mitigate the risk of overloading or blackouts in the energy grid Abubakar, Khalid, Mustafa, Shareef, and Mustapha (2017); Bonfigli et al. (2018).

NILM assists users by providing appliance-level energy data. It involves breaking down the total power consumption of a building into individual appliance usages without installing sensors

on each device. Load identification, on the other hand, is the process of recognizing different appliances by analyzing their voltage and current signals. This step is fundamental for NILM, as it helps determine the number and types of appliances present, as illustrated in Figure 1.1. Additionally, load identification enables users to detect faulty appliances that consume excessive energy, which can impact costs and waste resources. These tasks can be addressed using either supervised or unsupervised methods. In supervised learning, machine learning models are trained with labeled power and signal data for NILM and load identification. Conversely, unsupervised methods use unlabeled data, requiring the models to learn patterns on their own through clustering or generative models. Supervised approaches typically achieve higher accuracy but depend on costly, labor-intensive labeled datasets, limiting scalability. Unsupervised approaches are more adaptable but often face lower accuracy and difficulties in accurately identifying appliances. A key challenge in NILM is differentiating between similar power signatures among various appliances, overlapping usage patterns, and changing power consumption behaviors over time, making precise disaggregation inherently difficult. Moreover, the absence of standardized and limited labeled datasets and the challenge of applying models across different households in real time present ongoing obstacles for both methods.

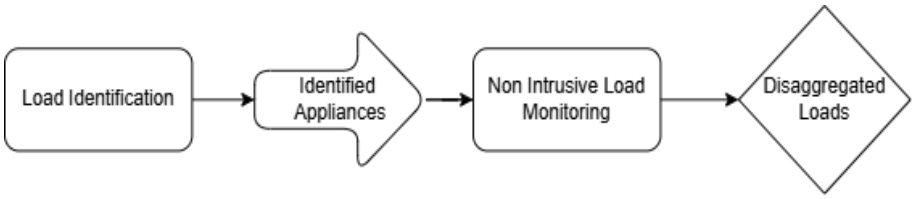

Figure 1.1: Flowchart of Load Identification and Non-Intrusive Load Monitoring

1.1 Problem Formulation

This section formulates the problem statement of NILM and load identification mathematically, highlighting the enhancements introduced to improve performance.

1.1.1 Non-Intrusive Load Monitoring

The goal of NILM is to disaggregate the total power consumption of a building into individual appliance-level power signals. Let the total power consumption at time t, denoted as $P(t)$, be the sum of the power consumed by N individual appliances as shown in equtation 1.

$$P(t) = \sum_{i=1}^{N} P_i(t) + \varepsilon(t) \tag{1}$$

where $P(t)$ is the total measured power at time t, $P_i(t)$ is the power consumed by the i^{th} appliance at time t, and $\epsilon(t)$ represents noise or measurement error. In a denoised scenario, the term $\epsilon(t)$ equals zero, whereas in a noisy scenario, $\epsilon(t)$ can include both measurement noise and interference from other appliances, such as unknown or always-on devices. The objective of NILM is to estimate $P_i(t)$ for each appliance i from the total power signal $P(t)$. This can be expressed as:

$$\hat{P}_i(t) = f(P(t), \theta) \tag{2}$$

where $\hat{P}_i(t)$ is the estimated power consumption of appliance i at time t, $f(\cdot)$ is a function (model) that disaggregates the total power $P(t)$ into individual components, and θ represents the parameters of the model.

1.1.2 Load Identification

The goal of Load Identification is to determine the state of the appliance (ON or OFF) by analyzing the energy data of the house's main supply lines (aggregate profiles). Every appliance has its own unique current and power, which can be used to generate distinct features such as current harmonics, reactive power, and apparent power. Whenever an appliance changes its status (OFF -> ON), it generates certain types of fluctuations in the current that alternatively cause changes in the features. These changes are then used to identify the loads Jiang, Wang, Qiu, Li, and Zhang (2025). Let, for each appliance i at time t, the voltage and current waveforms which are sampled over a

window of length L as shown in Equations (3-4).

$$\mathbf{v}_x(t) = [v_x(t_1), v_x(t_2), \ldots, v_x(t_L)] \in \mathbb{R}^L \tag{3}$$

$$\mathbf{i}_x(t) = [i_x(t_1), i_x(t_2), \ldots, i_x(t_L)] \in \mathbb{R}^L \tag{4}$$

where $x \in \{1, 2, \ldots, N\}$ is the appliance index, t is the time index, and it refers to a specific sampling window. Additionally, L refers to the number of samples in each waveform segment, and $v_x(t_k), i_x(t_k)$ are voltage and current samples for the appliance x, respectively. The feature extraction function $\phi(\cdot)$, which transforms the raw waveform current signal of the appliance x into a feature vector, can be defined as follows.

$$\mathbf{v}_x(t) = \phi\left(\mathbf{i}_x(t)\right) \in \mathbb{R}^d \tag{5}$$

$\phi(\cdot)$ may include statistical, frequency-domain, or time-domain features. $\mathbf{v}_x(t)$ is a feature vector for appliance i at time t and d is the dimensionality of the feature vector. This feature vector $\mathbf{v}_x(t)$ is then used to train the classification model f to map it to the ON/OFF probability of the appliance x as shown in Equation 6.

$$\hat{y}_i(t) = f\left(\mathbf{v}_x(t), \theta_i\right) \tag{6}$$

where $\hat{y}_i(t)$ are predicted appliances and θ_i is the parameter for model f.

1.2 Related Work

This section reviews recent and well-established research studies conducted in the domains of NILM and load identification. It highlights key developments, methodologies, and contributions that have shaped the current landscape of this field.

1.2.1 Non-Intrusive Load Monitoring

In recent research, Moreno et al. (2024) proposes two convolutional neural networks (CNNs): VGG16 and MobileNet. VGG16 is a well-known model recognized for its deep architecture and capacity to capture detailed spatial hierarchies, making it effective for feature extraction. In contrast, MobileNet is optimized for efficiency, employing depthwise separable convolutions to reduce computational demands while maintaining high accuracy. The paper introduces a weighted average confidence voting (WeCV) ensemble method, which combines predictions from both VGG16 and MobileNet to capitalize on their individual strengths, resulting in improved accuracy. In Yaniv and Beck (2024), the authors apply Robust Principal Component Analysis (RPCA) to reduce data dimensionality, isolating the most significant features for distinguishing between different electrical appliances, thereby enhancing classification with lower computational costs. In Chouchene, Amayri, and Bouguila (2024), sparse coding is used to develop a compact, efficient representation of energy consumption data, emphasizing key features. Transfer learning is also applied to leverage pre-existing knowledge from related tasks, enhancing model performance in identifying and isolating the energy usage of individual appliances from a single aggregate signal.

In Shang, Chen, Chen, and Lu (2024), a graph neural network is proposed to exploit the interconnected nature of household appliances by modeling their relationships as a graph. Initially, Gaussian random variables represent the graph edges, which are later refined based on observed appliance interrelationships, enabling simultaneous disaggregation of multiple appliances' energy consumption. Finally, Narges Zaeri Esfahani and Bahiraei (2024) presents a method for disaggregating total energy use in commercial buildings into primary end-uses like lighting, cooling, and heating. Using time series decomposition, this method separates energy data into components representing distinct usage patterns, validated against actual submetered data from ten office buildings in Ottawa, Canada, demonstrating its effectiveness in providing detailed energy insights without extensive submetering infrastructure. Most recent research in NILM focuses on optimizing the problem using novel methods and techniques.

1.2.2 Load Identification

In recent studies, Yan, Hao, Nardello, Brunelli, and Wen (2025) proposes a weighted transferable random forest (WTRF) model for generalizable load identification. WTRF uses transfer learning to adapt to new homes with only 1–3 labeled samples per appliance, updating a subset of decision trees via the Improved Structure Expansion/Reduction (ISER) algorithm. It is being tested on a Raspberry Pi 4 Board to evaluate its performance in an edge computing environment. Raspberry Pi has been widely used in most of NILM research because of its cost-effectiveness and support Kotsilitis, Marcoulaki, and Kalligeros (2024); Wu et al. (2023); Yan et al. (2025). In Mylona and Bouhouras (2025) the authors present a Digital Twin-based NILM framework that uses CNNs to classify appliance operation in real time from images of odd harmonic current distortions, specifically 3rd and 5th harmonic features. A VGG16-based CNN is used for classification, achieving high accuracy in detecting single, combined, and event-based appliance states. It outperforms existing models with fewer features and enables interactive monitoring through digital twin integration.

Lu et al. (2025) proposes a color-coded image-based load identification method that maps reactive power, power factor, and current sequence features to RGB channels, integrating harmonic information into mixed-color images. A lightweight model using Residual Shrink Building Unit with Channel-Shared Threshold (RSBU-CW) residual units is developed and achieves high accuracy across multiple datasets. Compared to ordinary color image methods, the mixed-color approach improves recognition accuracy. The method enhances smart grid reliability by enabling real-time monitoring and fault detection. Limitations include restricted load diversity and a lack of testing on higher-voltage systems.

In de Aguiar et al. (2025), the authors introduce a new high-frequency public dataset and a framework for jointly identifying electrical loads and photovoltaic (PV) distributed generation (DG) in NILM systems. The dataset includes voltage and current signals sampled at 1 kHz from residential appliances and a PV inverter. The authors test deep learning models, including InceptionTime, DeepDFML, and Sequencer, in three classification tasks: identifying only the inverter, only loads, and both together. Appliances include resistive (e.g., electric iron) and nonlinear loads (e.g., induction motor, drill + transformer, dimmer). The framework evaluates how DG presence affects

load classification and vice versa. The study shows that PV presence slightly affects nonlinear load classification but not inverter detection. A limitation is the small number of appliances and the need for more diverse real-world scenarios to improve generalization.

In Gao, Zhang, Wang, Tan, and Liang (2025) the authors proposed a model consisting of a feature extraction layer, a channel attention module, and a linear layer. Its process starts by converting the steady-state voltage and current signals from an electrical device into a colored V-I trajectory map. This map, which uses color to represent various electrical characteristics, serves as the input image for a CNN. The CNN automatically extracts spatial features from the image using its convolutional and pooling layers. Following this, the channel attention mechanism analyzes the extracted features. It generates weights for each feature channel by using global pooling and fully connected layers. This allows the model to adaptively adjust the contribution of different channels, effectively focusing on the most important features for identification. Finally, these re-weighted features are processed by the linear layer to classify the appliance.

1.3 Contributions

This research examines load disaggregation, focusing on NILM and its sub-task, Load Identification. The key contributions of this thesis are outlined as follows:

- **Feed Forward Neural Network for Non-Intrusive Load Monitoring:** In this study, we developed a NILM method based on a simple and effective Feedforward Neural Network (FFNN). The training data is oversampled to enhance the detection of appliance activation cycles. Additional features are incorporated by utilizing both aggregate profiles, improving model input diversity.

- **Appliance Identification Using Kolmogorov–Arnold Networks with Extended Feature Extraction and Saliency Analysis:** In this study we design a straightforward yet efficient Kolmogorov-Arnold Network (KAN) model for load identification. Approximately 75 distinct features across three categories have been extracted from the voltage and current data. Moreover, only effective features are selected for training the model by using effective feature

analysis. To evaluate the scalability of the approach, a substantial number of appliances from 3 different datasets have been used in the investigation to study the performance of the model as the number of appliances increases.

1.4 Thesis Overview

This thesis is structured into four chapters as follows:

- Chapter 1 presents the background and objectives of the research. It defines the problem statement, reviews existing theories and literature on load disaggregation, and highlights the key contributions of this thesis.
- Chapter 2 explores NILM for household energy consumption. It introduces a FFNN model enhanced with oversampling and feature amplification techniques to improve appliance disaggregation performance. The chapter concludes with an evaluation of the FFNN across various datasets and a discussion of the results.
- Chapter 3 introduces a new supervised load identification technique for recognizing individual appliances. Despite limited labeled data, the model demonstrates the ability to accurately predict various appliances. The proposed approach is trained and evaluated using three datasets. Additionally, the section explains the feature extraction process and discusses the importance of each extracted feature.
- Chapter 4 concludes the thesis by summarizing the key findings and main contributions of the research.

Chapter 2

Feed Forward Neural Network for Non-Intrusive Load Monitoring

2.1 Introduction

Energy disaggregation presents several critical challenges. One of the primary difficulties is the overlapping appliance signatures, where devices with similar energy consumption patterns generate nearly indistinguishable load profiles, making it hard to differentiate between them. Furthermore, the detection of low-power appliances becomes even more challenging when their activations overlap with high-power appliances, as the larger consumption masks the smaller. External noise and signal interference further complicate the analysis, distorting the data and making it harder for models to classify appliance usage accurately. Additionally, many appliances operate in multiple states, adding complexity to their identification since their transient and steady-state behaviors vary significantly. Generalization is another key issue; models trained on data from one household often struggle when applied to new households with different sets of appliances and usage patterns. Moreover, ensuring real-time energy data processing while maintaining privacy is a major concern, as is the challenge of achieving cost-effective deployment in practical, real-world scenarios.

From an algorithmic perspective, a well-designed privacy-protected energy disaggregation model for NILM should meet the following criteria:

- Achieve high accuracy.
- Perform consistently in the presence of noise, varying appliance behavior (different operational modes), or incomplete data (partial activations).
- Be computationally efficient, utilizing minimal resources during operation.

Most existing models tend to excel in one of these areas while compromising on the others. Owing to their remarkable performance in disaggregation tasks, machine learning techniques have been extensively utilized. For instance, the Hidden Markov Model (HMM) is employed in Makonin, Popowich, Bajić, Gill, and Bartram (2016), and the multi-sequential, non-flush factorial hidden Markov model (MN-FHMM) in Liang and Ma (2020), along with various other HMM variations. For a comprehensive overview of recent studies, refer to Angelis, Timplalexis, Krinidis, Ioannidis, and Tzovaras (2022). Additionally, solutions leveraging Support Vector Machines (SVM) Figueiredo, de Almeida, and Ribeiro (2012), Decision Trees Gillis, Alshareef, and Morsi (2016), k-Nearest Neighbor (KNN) Figueiredo et al. (2012), and numerous other machine learning models have been proposed.

Deep Neural Networks (DNNs) have attracted considerable attention in recent years, as they have demonstrated exceptional performance in load disaggregation. In Nie, Yang, and Xu (2022), the authors proposed an encoder-decoder model, where the encoder transforms the input into an embedding matrix and feeds it into a residual neural network (ResNet50). The decoder employs a Transformer architecture with multiple attention mechanisms, which increases its complexity but yields promising results. In the same study, the authors discuss the benefits and limitations of several deep learning models. In contrast, the authors in Bonfigli et al. (2018) introduced a Denoising Auto-Encoder-Decoder model, which is more lightweight. This model incorporates convolutional, max-pooling, up-sampling, and dense layers, along with an early-stopping criterion to prevent overfitting. Additionally, dropout layers were employed. Though the overall results are satisfactory, the model underperforms significantly for certain appliances. In Y. Liu, Liu, Shen, Zhao, and Gao (2021) the authors proposed a Deep Dictionary model in which, first, an adaptive window-based detection technique is used to manage different types of overlapping combinations and detect state changes. Next, a deep dictionary learning model is proposed for real-time load monitoring. Lastly, a

sparse coding algorithm is formulated to address the simultaneous occurrence of multiple switching events effectively. This model performs well in handling the overlap of various electrical appliances. However, it has a limitation in that it struggles to identify unknown appliances.

In Rafiq, Manandhar, Rodriguez-Ubinas, Ahmed Qureshi, and Palpanas (2024), the authors compare deep learning, machine learning (ML), and advanced machine learning models comprehensively. The paper highlights that traditional ML models, such as Support Vector Machine (SVM) Altrabalsi, Liao, Stankovic, and Stankovic (2014), Artificial Neural Network (ANN), K-Nearest Neighbors (K-NN) Altrabalsi et al. (2014), Naive Bayes and Decision Tress (DT) Gillis et al. (2016), are relatively easy to implement and require significantly fewer computational resources compared to deep learning models. However, this simplicity comes at the expense of reduced accuracy. On the other hand, deep learning models including Recurrent Neural Networks (RNN) Linh and Arboleya (2019), Long Short-Term Memory (LSTM) Song et al. (2021), Bi-directional LSTM (Bi-LSTM) Rafiq, Shi, Zhang, Li, and Ochani (2020), Convolutional Neural Networks (CNN) Athanasiadis, Doukas, Papadopoulos, and Chrysopoulos (2021), Autoencoders Massidda, Marrocu, and Manca (2020), and Gated Recurrent Units (GRU) Kalinke, Bielski, Singh, Fouché, and Böhm (2021) offer superior accuracy and, when optimized, perform exceptionally well in disaggregation tasks. Despite their effectiveness, these models require substantial computational power and vast labeled datasets for training, which can be limiting in practical scenarios. To address these challenges, we proposed a deep learning feed-forward neural network model that is both computationally efficient and capable of delivering good or acceptable results. By introducing novel features and employing oversampling techniques, we were able to enhance the model's performance without sacrificing computational efficiency. The operating characteristics of the appliance include harmonics, active power, voltage, current, and V-I trajectory Nie et al. (2022); Wang, Chen, Guo, and Xu (2021). The majority of contemporary research is centered on active power Akbar et al. (2024); Athanasiadis et al. (2021); Bonfigli et al. (2018); Bucci et al. (2020); Kalinke et al. (2021); Linh and Arboleya (2019); Massidda et al. (2020); Nie et al. (2022); Rafiq et al. (2020); Song et al. (2021); Todic, Stankovic, and Stankovic (2023). In this study, we are also using active power for disaggregation.

2.2 Methodology

For pattern recognition and time-series data processing, Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs) are very efficient, but they require a lot of computational resources. We propose a simpler yet effective Feed-Forward Neural Network (FFNN) model to overcome this constraint. We increased the number of hidden layers, the number of neurons in each hidden layer, and the number of input features in order to improve accuracy and detection of appliance activation patterns. To ensure reliable model performance, the training data was also oversampled to account for varying appliance activation patterns. This approach improves accuracy across different appliances, with modest gains for some and substantial improvement for others while maintaining low computational demands. Figure 2.1 outlines the workflow of the proposed NILM method based on a Feed Forward Neural Network (FFNN) model. Energy consumption data are initially gathered from households across three regions using the RAE Makonin (2017), REDD Kolter and Johnson (2011), and REFIT Murray, Stankovic, and Stankovic (2017) datasets. This data then undergoes preprocessing to ensure quality and consistency. During preprocessing, outliers are removed to allow for uniform normalization, and two scenarios Noised and Denoised are created for relevant datasets. After preprocessing, the data is split into training, validation, and testing sets. To enhance model performance, particularly for detecting appliance activation cycles, oversampling is applied to the training and validation sets. In the oversampling process, the refrigerator is used as a reference appliance to compensate for the low activation frequency of appliances like washers, dryers, dishwashers, and electric heaters. The FFNN model is then trained on the oversampled training data and fine-tuned with the oversampled validation set. Once trained, the model is investigated on the test data to generate the final disaggregation output. Seven different evaluation metrics are used to assess the model's accuracy, providing a comprehensive analysis of its performance. Lastly, a comparative analysis is conducted on washers from the three different regions. This analysis reveals that appliances from various regions or brands exhibit distinct activation patterns, which leads to varying model behavior for the same appliance across different datasets.

This section begins by presenting the architecture of our proposed approach, a Feed Forward Neural Network (FFNN) model. Following this, we describe the architecture of the benchmark

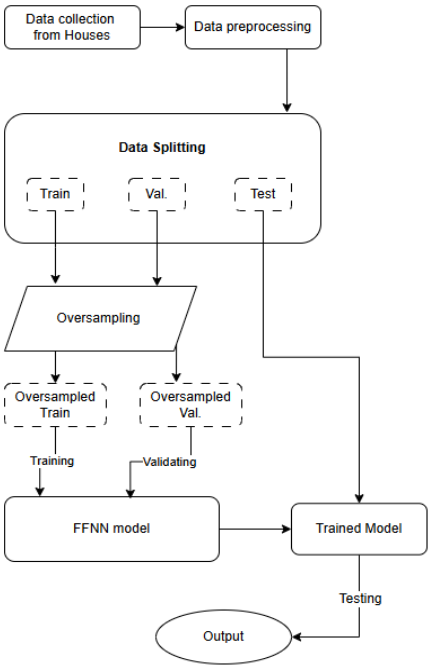

Figure 2.1: Flowchart of the whole study. where Train, Val., and Test are training, validation, and test sets respectively

approach, a Denoising Auto-encoder (DAE) from Bonfigli et al. (2018), used as a baseline for comparison. A comparative analysis is conducted to evaluate the accuracy and effectiveness of our model against the DAE model. Finally, we discuss the data preprocessing steps and the oversampling technique applied to enhance model performance.

2.2.1 Proposed Approach

FFNN is an artificial neural network where information flows in a single direction from the input layer, through one or more hidden layers, to the output layer. Each neuron in a layer receives input from the previous layer, processes it with weights and an activation function, and passes the result forward. No feedback connections exist, meaning data does not cycle through the network.

Our network comprises 11 layers, including an input layer, an output layer, and 9 hidden layers. Each layer employs the ReLU activation function, except for the output layer. ReLU is a linear function that outputs the input value when it is positive, and zero otherwise, effectively preventing negative values in the disaggregated active power. The parameters of the network are provided in

Table 2.1. We increased both the number of layers and neurons per layer to enhance the model's capacity for pattern recognition. However, the current configuration represents the maximum number of neurons and layers that provided satisfactory results. Further increasing the layers and neurons did not improve the model's performance and could negatively impact computational efficiency.

Table 2.1: Parameters for FFNN

Layer	**Number of Neurons**	**Activation Function**
Input	2	ReLU
Hidden 1	10	ReLU
Hidden 2	30	ReLU
Hidden 3	40	ReLU
Hidden 4	50	ReLU
Hidden 5	60	ReLU
Hidden 6	70	ReLU
Hidden 7	80	ReLU
Hidden 8	90	ReLU
Hidden 9	100	ReLU
Output	1	-

2.2.2 Benchmark Approach

In Bonfigli et al. (2018), the authors address the NILM (Non-Intrusive Load Monitoring) problem as a denoising challenge and propose a Denoising Auto-Encoder (DAE) model. The model architecture is structured as follows: the encoder consists of one or more convolutional layers to generate feature maps, each utilizing a linear activation function. Each convolutional layer is followed by max-pooling layers, with additional convolutional and pooling layers to further process the data. Before the decoder, the model incorporates one or more fully connected layers with ReLU activation. The decoder mirrors the encoder, but max-pooling layers are replaced with upsampling layers.

The model is optimized by minimizing the mean square error, and training is conducted using Stochastic Gradient Descent (SGD) with Nesterov momentum Sutskever, Martens, Dahl, and Hinton (2013), along with early stopping to prevent overfitting. During the disaggregation phase, overlapping signal components are combined using a median filter applied in a sliding window analysis of the aggregated power data.

2.2.3 Data Preprocessing

Numerous anomalies (shown in figure 2.2) were identified in the appliance-level data during the REFIT dataset's preprocessing phase. These outliers were removed before training, as their enormous values could skew the mean and variance, leading to complications during normalization. In real-world data, noise and recording errors are common across most domains. This is particularly true in energy disaggregation, where researchers often create denoised scenarios for simplicity and to assess model performance accurately. Following this approach, we propose two scenarios for the REFIT dataset: a noised and a denoised scenario for each appliance. In the noised scenario, as utilized in Bonfigli et al. (2018), the aggregate profiles include contributions from unknown appliances not explicitly recorded in the dataset. Noise is further introduced as different appliances intermittently connect and disconnect from power outlets.

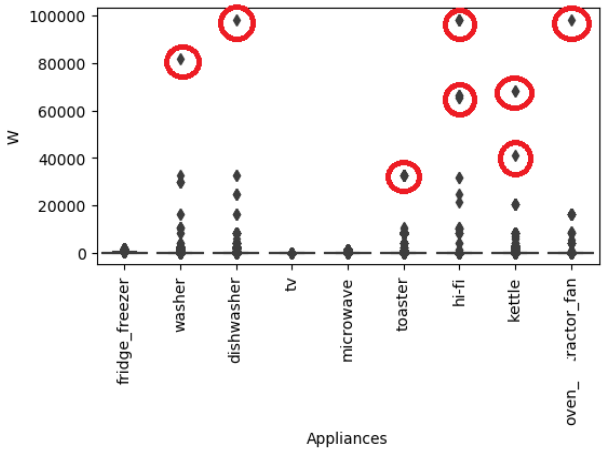

Figure 2.2: Box plot for House 2 REFIT dataset. Anomalies (outliers) are marked with red circles. Appliances are displayed on the x-axis. The y-axis represents the power in watts.

In the denoised scenario, in Bonfigli et al. (2018), the aggregate signal consists of the summed disaggregated power profiles of each appliance. For the denoised scenario, we employ an alternative approach where the aggregate profile is calculated as the sum of all appliances recorded in the dataset, rather than only summing the power profiles of appliances being disaggregated at specific time intervals. The REDD and RAE datasets are used without preprocessing since they contain two mains (aggregate power profiles), with some appliances connected to one main and others to both. The RAE dataset's aggregate power profiles are already denoised. However, for the REDD dataset,

the lack of a detailed circuit diagram prevents us from identifying which appliances are connected to each main, particularly for those with lower power consumption.

2.2.4 Oversampling

In household electrical appliances, a transient state occurs briefly when the appliance is first powered on or off, leading to fluctuations, whereas a stable state is reached when it operates at a steady power level. Appliances can be grouped into two categories: those that activate frequently at set intervals, like refrigerators, or are often manually operated, like microwaves; and those that activate less often, such as washers, dryers, electric heaters, and dishwashers. Appliances like washers, dryers, and dishwashers undergo multiple transitions between transient and stable states in each cycle. In contrast, refrigerators and microwaves generally shift states only once at the beginning and once at the end of their activation cycle.

For oversampling purposes, we use the refrigerator as a reference appliance due to its approximately 50% activation and 50% sparsity distribution. In contrast, data for less frequently used appliances is over 95% sparse, making activation detection challenging for the model. To address this, around 30% of the training data for each appliance is oversampled based on activations already present in the training and validation sets, while test data activations remain untouched to ensure unbiased evaluation on unseen data. The training data is not oversampled to precisely match the activation percentage of the reference appliance; instead, it aims to maintain a balance, enabling accurate predictions for both activations and sparse regions, which are prevalent in the test set. Oversampling criteria are uniformly applied across datasets for appliances such as washers, dryers, dishwashers, and electric heaters.

2.3 Experiments

This section presents the experimental evaluation conducted on the selected datasets and appliances. A comprehensive performance comparison of the proposed model is performed using the specified metrics. First, we describe our selected datasets. The experimental setup and procedures

are then outlined. Finally, the results obtained from these experiments are discussed in detail.

2.3.1 Data sets

Three public datasets were selected to evaluate our model across different scenarios. The first is the Rainforest Automation Energy (RAE) dataset Makonin (2017), which contains 1 Hz data (mains and submeters) for two houses in Canada, spanning 72 days for House 1 and 59 days for House 2. We conducted experiments using only House 2, as it includes readings from both main lines (main1 and main2). Appliances chosen for testing include the refrigerator, dishwasher, washer, and dryer. The second dataset, the Reference Energy Disaggregation Dataset (REDD) Kolter and Johnson (2011), provides data for six houses in the United States over 119 days, covering 92 appliances. The mains (aggregate profiles) data are sampled at 1 second, while the appliances are sampled every 3 seconds. We selected four appliances for our experiments: the dishwasher, microwave, refrigerator, and washer. The third dataset, REFIT Murray et al. (2017), consists of 20 houses in the UK and records 117 appliances over two years. We focused on one house for simplicity and comparison and conducted experiments on the refrigerator, washer, microwave, dishwasher, and electric heater. The characteristics of the datasets used in this study are presented in Table 2.2.

Table 2.2: Characteristics of the aforementioned datasets

Dataset	Release Date	Buildings	Total Appliances	Period	Characteristics	Aggregate Sampling	Appliance Sampling
RAE	2017	2	40	72 days for House 1 59 days for House 2	P,V,I	1 sec	1 sec
REDD	2011	6	92	119 days	P,V,I	1 sec	3 sec
REFIT	2016	20	177	2 years	P	8 sec	8 sec

There are two main reasons for selecting these datasets. First, the REDD and RAE datasets contain readings from both main lines (the main electrical lines for the house), improving the model's learning capabilities, as discussed in the experimentation section. Secondly, by choosing datasets from three different regions, we aimed to test our model on various types of appliances, ensuring its generalizability across different settings.

The selected houses from each dataset were divided into three distinct subsets: training, validation, and testing. Based on the true activations present in the data before oversampling, 70% of the data was allocated for training, which included the oversampled activations (detailed in the

"Experiments" subsection on oversampling). At the same time, 15% was assigned for validation and 15% for testing (without oversampling). Importantly, only unseen data that was not used during training was utilized for testing. The original dimensions of the data remained unchanged and were used as-is for both training and prediction tasks.

2.3.2 Evaluation metrics

For the evaluation and comparative analysis of our model, the following metrics have been selected.

R^2score

R^2 (R-squared) is a statistical measure that represents the proportion of the variance for a dependent variable (also known as the "response variable") an independent variable or variables in a regression model explain that. It provides insight into how well the independent variables predict or explain the variation in the dependent variable.

The formula for calculating R^2 is:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

where:

- SS_{res} is the sum of squares of residuals (the difference between the observed and predicted values):
$$SS_{\text{res}} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$
- SS_{tot} is the total sum of squares (the sum of the observed data):
$$SS_{\text{tot}} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$
where $\bar{y}$ is the mean of the observed data, y_i is observed value and $\hat{y}_i$ is the predicted value.

Values of R^2 range from 0 to 1, where an R^2 of 0 means the independent variables explain none

of the variation in the dependent variable and 1 means the independent variables explain all of the variation in the dependent variable.

Precision, Recall, NEP, and F1

Precision measures the proportion of power predicted for an appliance correctly assigned to it out of the total predicted power. **Recall** measures the proportion of actual power used by an appliance that the model successfully identifies or assigns correctly Bonfigli et al. (2018).

$$P_i = \frac{\sum_{t=1}^{T} \min(\hat{y}_i(t), y_i(t))}{\sum_{t=1}^{T} \hat{y}_i(t)}$$

$$R_i = \frac{\sum_{t=1}^{T} \min(\hat{y}_i(t), y_i(t))}{\sum_{t=1}^{T} y_i(t)}$$

The **Normalized Error in Assigned Power (NEP)** quantifies the difference between the estimated power $\hat{y}_i(t)$ and the actual power $y_i(t)$, normalized by the appliance's total energy usage Bonfigli et al. (2018). NEP is computed for each appliance i as follows:

$$NEP_i = \frac{\sum_{t=1}^{T} |y_i(t) - \hat{y}_i(t)|}{\sum_{t=1}^{T} y_i(t)}$$

where $\hat{y}_i(t)$ is the predicted value,$y_i(t)$ is the actual value of sample i, and T is the total number of samples.

F1-score is the harmonic mean of Precision and Recall, providing a balanced measure between the two:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Mean Absolute Error

The **Mean Absolute Error (MAE)** measures the average absolute difference between predicted and actual values in a dataset.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the total number of observations.

Mean Square Error

The **Mean Squared Error (MSE)** calculates the average squared difference between predicted and actual values, placing more weight on larger errors.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the total number of observations.

2.3.3 Experimental Setup

The model was trained using the following parameters: data was fed into the model in mini-batches with sizes ranging from 16 to 128. On the input data, a mean and variance normalization is calculated. To ensure uniform normalization throughout the dataset, a random training set sample is used to compute the mean and variance values. On the other hand, the maximum power consumption value of the associated appliance is used to conduct a min-max normalization on the target data Bonfigli et al. (2018). As previously mentioned, stochastic Gradient Descent (SGD) with a Nesterov momentum of 0.9 was employed for optimization. The initial learning rate was set to 0.001, which decreased by a factor of 10 if there was no improvement in the loss for up to 15 epochs. The minimum learning rate was capped at 10^{-6} (0.000001). Training was performed for a minimum of 50 epochs and a maximum of 100 epochs. Although we tested with more than 100 epochs, the model consistently converged within this range for all appliances. The duration of each experiment varied based on the size of the training dataset. Since the REDD dataset is approximately half the size of

the REFIT and RAE datasets, it requires less time to process. On average, each experiment took approximately 30 to 50 minutes to complete. The neural network was implemented using TensorFlow, with Scikit-learn Pedregosa et al. (2011) used for metrics such as R^2 score, MSE, and MAE, and Matplotlib Hunter (2007) and Seaborn Waskom (2021) for visualization. The experiments were conducted on an Intel Xeon processor (3.60 GHz), with 32GB RAM.

2.3.4 Training

During training, Stochastic Gradient Descent (SGD) with Nesterov momentum Sutskever et al. (2013) was employed. The data was fed into the model in mini-batches. The model was optimized to minimize the Mean Squared Error (MSE). If the MSE did not decrease for a specific number of epochs, the learning rate was reduced to ensure continued progress. A dropout layer with a rate of 0.2 was incorporated into the model to mitigate overfitting, particularly for certain appliances.

2.3.5 Results

This section will provide a detailed analysis of the model's performance on individual appliances. First, we examine the results for each appliance across different datasets. Then, we compare the results of the same appliance across datasets to explore performance variations. In addition, we will compare the performance of our model with the DAE model proposed in Bonfigli et al. (2018) on both the RAE and REDD datasets. This comparison allowed us to evaluate the robustness and generalizability of our model across different datasets. By using energy-based metrics such as F1 score and NEP, we assessed how effectively the FFNN model disaggregated appliance-specific energy consumption.

Table 2.3: Disaggregation performance on RAE dataset appliances.

Appliances	R^2 on Training Data	R^2 on Validation Data	R^2 on Test Data	MSE for Training Data	MSE for Test Data	Mean Absolute Error
Dishwasher	0.96	0.97	0.68	0.004914	0.001091	0.02242
Dryer	0.90	0.90	0.84	0.002099	0.00203	0.01089
Refrigerator	0.61	0.56	0.07	0.000346	0.000835	0.01673
Washer	0.67	0.51	0.143	0.001827	0.002438	0.01927

Table 2.4: Performance comparison of DAE and FFNN model on RAE dataset

Algorithm	Metric	Dishwasher	Dryer L1	Dryer L2	Refrigerator	Washer	Overall
DAE	F1(%)	49.8	91.2	-	39.1	11.9	48
Bonfigli et al. (2018)	NEP	0.64	0.131	-	0.94	4.416	1.53
FFNN	F1(%)	60	83	66	53	41	60.6
	NEP	1.022	0.35	0.66	0.75	1.44	0.84

Tables 2.3 and 2.4 summarize the results for appliances in the RAE dataset. The FFNN model performs best on high-energy-consuming appliances such as dishwashers and dryers. These appliances exhibit significant changes in the aggregate energy profiles, making it easier for the model to detect their activation. Figure 2.3 presents the correlation matrix between the aggregate energy profiles (mains) and specific appliances, including the dryer and refrigerator, providing insight into the relationship and interaction between these energy profiles. Where l1 and l2 are aggregate profiles. The dryer is connected to both main lines and consumes a substantial amount of energy, exhibiting a strong correlation with both aggregate profiles. In contrast, the refrigerator, connected to line L2, shows a weaker correlation with the aggregate profiles due to its comparatively lower energy consumption. To improve detection accuracy, we applied oversampling techniques to these appliances, incorporating different types of activation sequences. Due to recent technological advancements, many appliances now possess multiple features, leading to variations in energy consumption patterns and activation durations.

In contrast, appliances like refrigerators and more specifically washers in this dataset consume considerably less energy than dishwashers and dryers. After evaluating results across datasets, we will conduct a comparative analysis for the washer. Larger appliances, such as clothes dryers, often utilize two electrical lines (L1 and L2) in Canadian households Makonin (2017). The RAE dataset contains readings from two aggregate energy profiles. For simplification, many energy disaggregation studies combine these profiles; however, in this paper, we improve model performance by using the aggregate profiles without combining them. As shown in Figure 2.4, the dryer is connected to both L1 and L2. To handle this, we employed multi-variable regression, and Tables 2.3 and 2.4 present the results for both connected profiles (L1 and L2).

From Table 2.3, the R^2 value close to 1 for the dishwasher indicates that the regression line provides an excellent approximation of the actual data, reflecting a strong fit between the model's

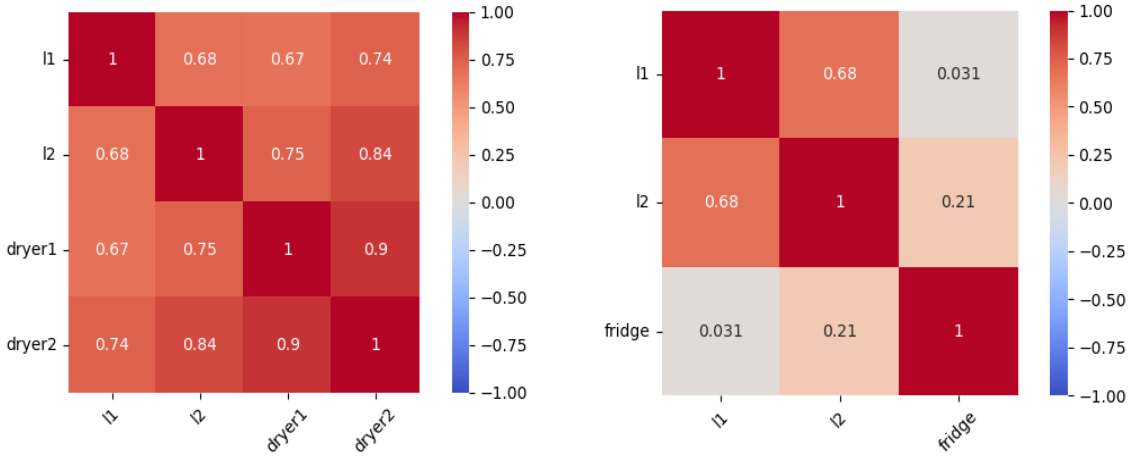

Figure 2.3: Correlation matrix for dryer and refrigerator from the Rae dataset.

Phase Log	Main ID	Sub ID	Breaker	Label	Meter# (CT)
L1	17	B	30A	Clothes Dryer	B.1 (50A)
		A			
L2	15	B	30A		B.2 (50A)
		A			

L2	100A	Line in from Garage Main Panel	A.2 (100A)
L1	100A	Line in from Garage Main Panel	A.1 (100A)

Figure 2.4: Part of the panel of house 2 from RAE dataset Makonin (2017) shows that the dryer is connected to both mains (l1 & l2). The whole panel is present in the RAE dataset

predictions and observed values, but it drops slightly for the test data. Due to the limited number of dishwasher activations in the test data, R^2 is decreased for the test data. Additionally, the F1 score is lower than expected given the high R^2 values for training and validation. This slight performance decline in the test data is due to variations in activation modes not present in the training set, as well as overlapping activations from other appliances. The latter issue leads to false positives, thereby increasing the NEP. The refrigerator, in contrast, has the lowest R^2 for both the training and test datasets. Despite consuming a substantial portion of the total household energy, the refrigerator's frequent, low-energy activation cycles typically lasting 15 to 20 minutes, make it particularly difficult to disaggregate. Figure 2.5. displays the total energy consumption for house 2 from the RAE dataset, detailing the energy usage across individual appliances. The refrigerator consumes 15.9% of the total energy, while the clothes dryer and HVAC boiler also account for a substantial portion,

often overlapping with the refrigerator's usage. Plugs and lights contribute about half of the total energy, introducing significant noise into the data, as these appliances are included in the aggregate energy profiles but are not individually disaggregated. These two factors contribute to poor disaggregation accuracy for appliances such as refrigerator and washer.

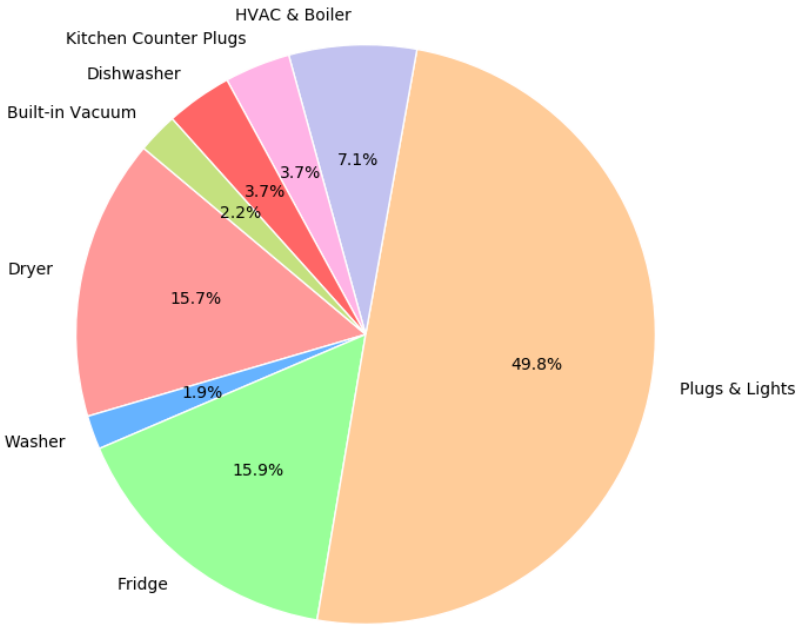

Figure 2.5: Percentages of energy consumed (in kWh) over the 59 days for RAE dataset appliances.

Compared to the DAE model, the FFNN model shows significant improvements and some trade-offs for different appliances. The individual appliance performance comparison between the FFNN and DAE models for RAE dataset appliances is as follows: For the dishwasher, the absolute improvement in the F1 score is **10.2%**, though the Normalized Error in Assigned Power (NEP) shows the increase of **0.38**. For the refrigerator, there is an **13.9%** increase in the F1 score, with a slight reduction of **0.19** in NEP. For the washer, the F1 score improves substantially by **29.1%**, accompanied by **0.66** decrease in NEP. For the dryer, a direct comparison was not feasible. In Bonfigli et al. (2018), the authors used amalgamated energy profiles of the dryer from AMPds dataset Makonin, Ellert, Bajic, and Popowich (2016) (representing data from the same household as the RAE dataset but with a different sampling frequency), whereas our approach treats them separately. Consequently, our model provides two distinct predictions corresponding to each energy consumption

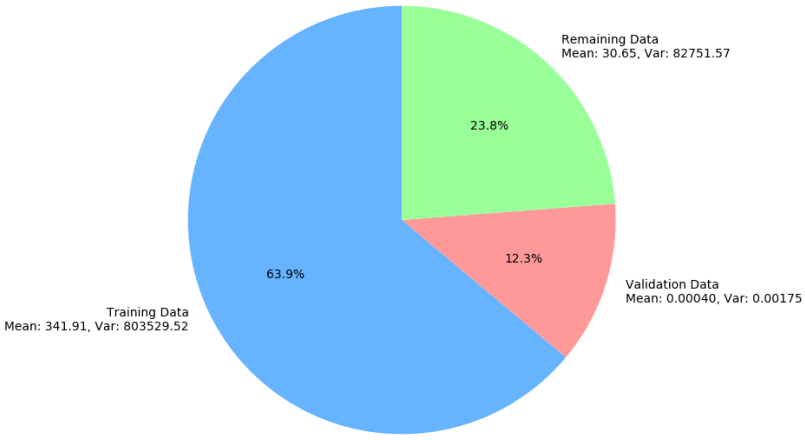

Figure 2.6: Distribution of data for washer of REDD dataset before the oversampling of validation data

line for the dryer. By taking the mean of both predictions, the F1 score decreases by **16.7%** and NEP is increased by **0.37**. Overall, the FFNN model outperforms the DAE model in general performance by **12.6%** increase in F1 score and **0.69** decrease in NEP. In terms of appliance-specific results, the FFNN model performs better than the DAE model for all appliances except for dryer.

The results for the REDD dataset are presented in Tables 2.5 and 2.6. The FFNN model exhibited the best performance on the appliances in the REDD dataset. Unlike the RAE dataset, the energy activation distributions for each appliance in the REDD dataset are uneven. This imbalance results in a significant discrepancy between the R^2 values for the training and validation data. Before oversampling the validation data of the washer, the washer displayed an extremely negative R^2 value for validation and 0.96 for training. This large difference is attributed to the substantial variation in training and validation data distribution. Figure 2.6 illustrates the validation, training, and remaining data distribution for the washer. For the training data, the mean is 341.58 with a variance of 802,874.53, whereas for the validation data, the mean is 0.0004 and the variance is 0.0025. This disparity negatively impacted data normalization, leading to poor R^2 performance on the validation set. To address this issue, we oversampled the validation data using activation sequences from the training data. The R^2 value for the validation data in Table 2.5 represents the model performance with oversampling of validation data.

Additionally, we used the microwave as an appliance for disaggregation from the REDD dataset.

Table 2.5: Disaggregation performance on REDD dataset appliances

Appliances	R^2 on Training Data	R^2 on Validation Data	R^2 on Test Data	MSE for Training Data	MSE for Test Data	Mean Absolute Error
Dishwasher	0.94	0.04	0.87	0.00199	0.00255	0.01115
Microwave	0.69	0.34	0.45	0.00107	0.00576	0.01002
Refrigerator	0.80	0.14	0.61	0.00022	0.00073	0.01006
Washer	0.97	0.97	0.77	0.00261	0.00404	0.01050

Table 2.6: Performance comparison of DAE and FFNN model on REDD dataset

Algorithm	Metric	Dishwasher	Washer	Refrigerator	Microwave	Overall
DAE	F1(%)	41.8	-	60.4	13.6	38.6
Bonfigli et al. (2018)	NEP	0.756	-	1.053	1.752	1.187
FFNN	F1(%)	80	78	79	47	71
	NEP	0.4	0.56	0.37	0.76	0.523

Table 2.7: Disaggregation performance on REFIT dataset (Denoised scenario)

Appliances	R^2 on Training Data	R^2 on Validation Data	R^2 on Test Data	MSE for Training Data	MSE for Test Data	Mean Absolute Error
Electric Heater	0.92	0.65	0.81	0.00356	0.00251	0.0079
Dishwasher	0.76	0.20	0.76	0.00024	0.00039	0.00287
Refrigerator	0.71	0.71	0.69	0.00022	0.00022	0.00538
Microwave	0.86	0.91	0.13	0.00032	0.00073	0.0017
Washer	0.67	0.51	0.14	0.00183	0.00244	0.01927

Table 2.6 shows that the microwave had the lowest F1 score and the highest NEP. Although the energy consumption of the microwave during its activations is higher than that of the refrigerator, making it detectable by the model, the microwave's varied functionalities introduce multiple types of activations in terms of energy consumption and duration. Each time the microwave is used, different time settings are chosen by the user, and many of its activations are of very short duration. These factors make the microwave one of the most challenging appliances to disaggregate. Due to these complexities, many models struggle with microwave disaggregation. For instance, in the noisy scenario, the DAE model exhibited a 13.6% F1 score for the microwave.

Table 2.6 presents the results of the DAE model, alongside the FFNN model's performance in noisy scenarios for the dishwasher, refrigerator, and microwave. The FFNN model consistently outperforms the DAE model for these appliances, achieving higher true positive rates, which results in a lower NEP. Even though the DAE model does not include results for the washer, the FFNN model still surpasses its overall performance, even when accounting for one additional appliance in the evaluation. The individual appliance performance comparison between the FFNN and DAE models

for REDD dataset appliances is as follows: For the dishwasher, FFNN yields a **38.2%** increase in F1 score and **0.36** decrease in NEP. The refrigerator also shows improvements, with F1 rising by **18.6%** and NEP reducing by **0.68**. Similarly, for the microwave, FFNN achieves a **33.4%** boost in F1 and **0.99** decrease in NEP. However, a comparison for the washer is unavailable, as the DAE model did not disaggregate this appliance. Still, FFNN model shows satisfactory performance on washer with **78%** F1 score and **0.56** NEP. Overall, the FFNN model outperforms the DAE model in general performance by **32.7%** increase in F1 score and **0.67** decrease in NEP. In terms of appliance-specific results, the FFNN model performs better than the DAE model for all the appliances.

Table 2.8: Disaggregation performance on REFIT dataset (Noised scenario)

Appliances	R^2 on Training Data	R^2 on Validation Data	R^2 on Test Data	MSE for Training Data	MSE for Test Data	Mean Absolute Error
Electric Heater	0.62	0.41	0.36	0.01664	0.0176	0.05014
Dishwasher	0.44	0.77	0.17	0.00471	0.00891	0.01868
Refrigerator	0.38	0.37	0.36	0.00047	0.00046	0.01451
Microwave	0.29	0.15	0.05	0.00672	0.0008	0.00261
Washer	0.12	0.04	-0.01	0.0033	0.00511	0.01611

Table 2.9: F1 score and NEP on REFIT dataset in denoised scenario

Appliance	F1(%)	NEP
Electric Heater	87	0.28
Refrigerator	86	0.28
Dishwasher	64	1.02
Washer	41	1.44
Microwave	42	1.84

Similar to the oversampling performed on the REDD dataset's validation data, oversampling was also applied to the validation data for REFIT dataset appliances. The results of the disaggregation experiments on denoised aggregate profiles for REFIT dataset appliances are presented in Tables 2.7 and 2.9. In addition to the refrigerator, microwave, dishwasher, and washer from the REFIT dataset, we included an electric heater for disaggregation analysis. Due to its consistently high energy consumption, the electric heater achieved the most accurate disaggregation performance across all tested appliances and datasets. This superior performance is attributed to its strong correlation with the aggregate energy profile. The washer's test data contains limited activations, and the microwave exhibits similar behavior to what was observed in the REDD dataset; as a result, both appliances yield the lowest R^2 scores on the test data, which subsequently leads to the lowest F1

scores.

Table 2.10: F1 score and NEP on REFIT dataset in noised scenario

Appliance	F1(%)	NEP
Electric Heater	40	1.8
Refrigerator	64	0.74
Dishwasher	36	1.53
Washer	10	2.44
Microwave	15	2.82

The results for experiments using the noised aggregate profile from the REFIT dataset are shown in Tables 2.8 and 2.10. In the noisy scenario, the FFNN model achieves its highest accuracy with the refrigerator; however, its performance deteriorates for other appliances. The substantial noise within the REFIT dataset introduces frequent false positives and false negatives, leading to elevated NEP values. Additionally, as noted earlier, the limitation of having only a single feature in the REFIT dataset further constrains the model's disaggregation capability, adversely affecting its overall performance. The FFNN model performs better on the denoised aggregate profile for all appliances.

Figure 2.7 illustrates the activation cycles for the washer across each dataset. Specifically, in the RAE dataset, the model's performance on the washer is suboptimal not only for the FFNN model but also for the DAE model. This underperformance is due to the lower energy consumption of each washer cycle compared to that observed in the REDD and REFIT datasets. Additionally, each activation cycle in the RAE dataset varies significantly in pattern. Thus, appliances from different regions exhibit substantial differences in activation cycles, which impacts model performance. These variations may arise from different operational modes within the same appliance; however, the pronounced disparity in energy consumption suggests that other factors are at play. The highest F1 score for the washer is observed in the REDD dataset, attributed to its uniform activation pattern, the presence of two features (aggregate profiles), and higher energy consumption.

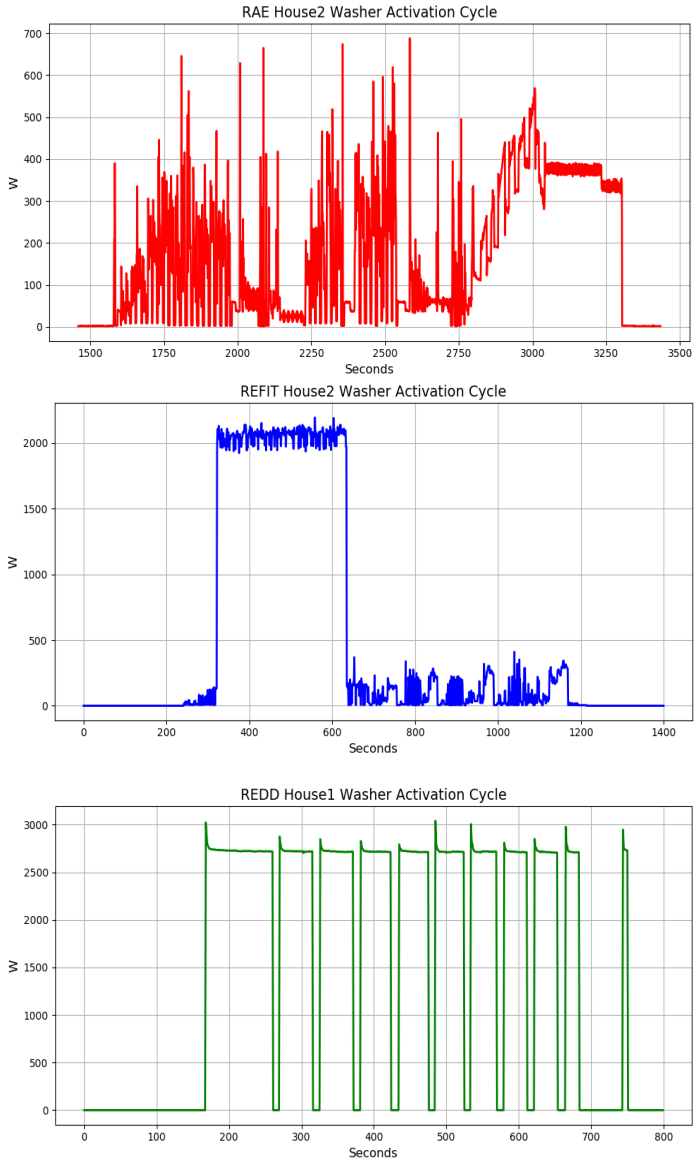

Figure 2.7: Activation cycles of the washer from RAE, REFIT, and REDD dataset

Chapter 3

Appliance Identification Using Kolmogorov–Arnold Networks with Extended Feature Extraction and Saliency Analysis

3.1 Introduction

Non-Intrusive Load Monitoring (NILM) methods are generally divided into two types: event-based and non-event-based. Non-event-based methods analyze the entire power signal over time, while event-based methods focus on sudden changes in overall power that suggest an appliance has been turned on or off. Identifying which appliance caused each change known as load identification which is a key part of event-based NILM and helps recognize the unique patterns of different devices Xiang et al. (2022). NILM breaks down total energy use into data for each appliance, making energy use more transparent and easier to manage. When load identification is accurate, it helps connect these changes to specific appliances, leading to better feedback, smarter energy use, and lower energy costs Kelly and Knottenbelt (2015).

There are two ways to identify the appliance status (on/off). The first approach considers individual sensors for each appliance. This solution is highly precise and accurate, but at the same time, it is too complex and expensive from a hardware perspective. On the other hand, the most practical approach involves detecting a signal at the main power bus and using a NILM algorithm to identify the appliance. This method simplifies hardware requirements but shifts complexity to signal processing through load identification techniques. Load identification poses several significant challenges. One of the primary issues is the occurrence of overlapping appliance signatures, where low-power-consuming appliances are activated concurrently with high-power-consuming appliances, masking the signal of the lower-power appliances. This overlap can hinder accurate detection and classification. Furthermore, external noise and signal interference increase the complexity of the problem by distorting measurement data, which in turn reduces the effectiveness of load identification models.

Another challenge arises from the presence of appliances with identical or similar operational characteristics, making it difficult to distinguish between them solely based on power consumption patterns or their voltage and current signals. In Hart (1992) the author categorized household electrical appliances into four distinct types based on their operational behavior:

- Type 1: Binary-state devices that operate in two modes (ON/OFF); e.g., table lamps.
- Type 2: Multistate devices or finite state machines (FSMs), such as washing machines and heat pumps.
- Type 3: Continuously variable devices (CVDs), characterized by their non-repetitive and variable power consumption patterns. An electric drill exemplifies this category Bucci et al. (2020).
- Type 4: Constant load or permanent consumer devices that operate continuously over extended periods (days or weeks), including smoke detectors, telephone sets, and cable television receivers Zeifman and Roth (2011).

Types 2 and 3 raise a new challenge in detecting multistate and continuously variable appliances, as these appliances may have different initial signals for different modes. Also, most of the

traditional datasets don't have multistate data which makes the model struggle to classify multistate appliances. Generalization remains a major challenge, as models trained on one household often underperform on others due to appliance and usage variability. Additionally, real-time processing, privacy preservation, and cost-effective deployment are critical concerns for practical NILM applications. From an algorithmic perspective, an ideal load identification model should demonstrate several key properties: it must achieve high accuracy, be computationally efficient, and maintain consistent performance under a variety of conditions. These conditions include the presence of similar appliances from different manufacturers, electrical noise interference, multistate operational behaviors, and overlapping power consumption patterns.

Most existing machine learning (ML) and Deep learning (DL) models tend to achieve high accuracy neglecting the computational cost of the models. For instance, in Xiang et al. (2022), the authors introduced a method involving the feature fusion of Power and current, converting it into color-coded 2D images. These features are then used as input to a CNN to classify images. First, changing the features into images is a computationally expensive step, and using CNN to classify them is more expensive especially when the number of appliances increases. Only a limited number of appliances are being used for investigation but day by day number of appliances is increasing in modern households. Most of the models struggle in that context. In Y. Liu, Wang, et al. (2021), the authors proposed a probabilistic ensemble model trained on a dictionary. Each column of the dictionary is treated as an atom, and a dictionary learning model is established through linear combinations. The idea is interesting but also computationally complex. Recently, most of the research has been based on engineered features like active/reactive power Hart (1992) and harmonics features Reinhardt, Burkhardt, Zaheer, and Steinmetz (2012); Srinivasan, Ng, and Liew (2006) rather than on dictionary-based training approaches. In Roos, Lane, Botha, and Hancke (1994), the authors conducted an in-depth study on steady-state appliance signatures to identify industrial electrical loads. However, their approach involves complex calculations to obtain precise power signature data. Moreover, creating a load model that can effectively classify and identify appliances under constantly fluctuating load conditions is still an open issue that needs further investigation.

To address these challenges, we propose the adoption of the Kolmogorov-Arnold Network (KAN) Z. Liu et al. (2025), which is a better alternative to multilayer perceptron (MLP). We have

also investigated the performance of Deep Neural Networks (DNNs) which shows that KAN is not only computationally less expensive than DNN but also much simpler. In contrast to most existing studies that primarily focus on features derived from voltage and current signals, our work also utilizes such features.

3.2 Methodology

For multiclass classification of appliances, CNNs and ensemble models such as Random Forest have been developed and shown to be highly effective. However, these models often require significant computational resources to achieve high accuracy. Also, most of the studies are focused on the algorithmic side of Load Identification problem. We propose the Kolmogorov-Arnold Network (KAN) as an efficient alternative to the traditional multilayer perceptron (MLP), while also emphasizing the importance of feature selection for appliance classification. A significantly smaller KAN can perform like the large CNNs and ensemble models in terms of accuracy. To the best of our knowledge, KAN has not yet been applied to load identification. Figure 3.1 outlines the workflow of the proposed model. Appliance signatures, including current and voltage waveforms, were collected from real households using the COOLL Picon et al. (2016), PLAID Medico et al. (2020), and WHITED Kahl, Haq, Kriechbaumer, and Jacobsen (2016) datasets. These signals were then used to extract 75 features. To minimize computational overhead and avoid large network architectures, features were distributed into different combinations. The data was then split into training (80%) and testing (20%) sets and fed into the KAN.

The model was trained and evaluated multiple times, and its average performance was calculated using four evaluation metrics. For each feature combination, this process was repeated to determine the most contributing features. Additionally, a correlation matrix was generated to identify and exclude redundant, highly correlated features. In the final phase, only the most informative and uncorrelated features were used to train the model. A comparative analysis was conducted between all the datasets to understand how an increasing number of appliances affects model efficiency. Finally, we discuss the limitations of KAN and propose directions for future research.

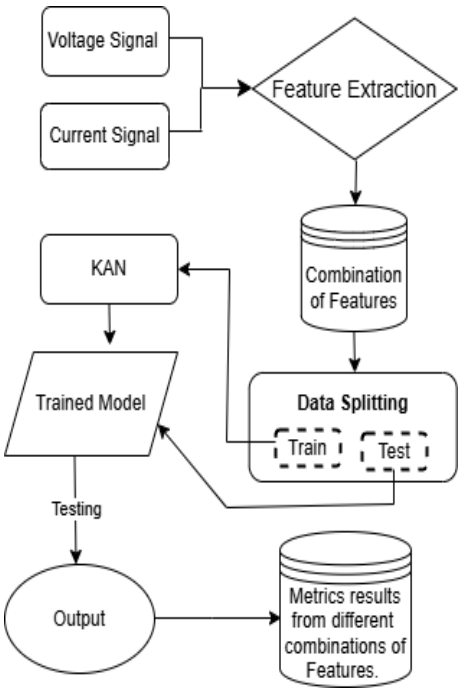

Figure 3.1: Flow chart of the whole study, where Train, Test are training and testing sets, respectively.

3.2.1 Proposed Approach

KAN is a type of neural network that applies learnable univariate functions to each input dimension in place of conventional weight-based linear transformations. Each node computes a trainable non-linear function of a single variable in a sequence of layers, after which the nodes combine linearly. Through functional learning, KANs actively modify the complexity and structure of the transformation, in contrast to conventional networks that only use matrix multiplications and preset activation functions. Standard activation functions are not necessary with this design, which enables enhanced accuracy and interpretability Z. Liu et al. (2025).

Table 3.1: Parameters for KAN

Layer	**Number of Neurons**
Input	Input Dimensions
Hidden 1	16
Hidden 2	32
Output	Num of classes

Our network comprises 4 layers, including an input layer, an output layer, and 2 hidden layers.

The parameters of the network are given in Table 3.1. The number of neurons in the input layer depends on the number of features used to train the KAN, and the number of neurons in the output layer depends on the number of appliances present in the dataset. Meanwhile, the number of neurons in the hidden layers varies from 16 to 32. The model's performance was not enhanced by adding more layers or neurons, negatively impacting the computing efficiency. The Kolmogorov–Arnold Network (KAN) used in our study applies three successive nonlinear transformations on the input vector, defined in Equation 7.

$$\hat{y} = \Phi^{(3)}\left(\Phi^{(2)}\left(\Phi^{(1)}(x)\right)\right) \tag{7}$$

Each layer-wise transformation $\Phi^{(l)}$ is computed as:

$$\Phi_j^{(l)}(x) = \sum_{i=1}^{n_l}\left(w_b^{(l)} \cdot \text{silu}(x_i) + w_s^{(l)} \cdot \sum_k c_k^{(l)} B_k(x_i)\right) \tag{8}$$

Where $x \in \mathbb{R}^{n_0}$ is the input feature vector, which may vary in dimensionality between tests. $\hat{y}$ is the final output of the network. $\Phi^{(1)}, \Phi^{(2)}, \Phi^{(3)}$ denote the transformations performed by the three KAN layers. $\Phi_j^{(l)}$ is the j-th output of layer l, computed by applying a nonlinear transformation over each input coordinate x_i. $\text{silu}(x) = \frac{x}{1+e^{-x}}$ is the base activation function. $B_k(x_i)$ are B-spline basis functions. $c_k^{(l)}$ are learnable spline coefficients for layer l. $w_b^{(l)}$ and $w_s^{(l)}$ are trainable scalar weights that control the contribution of the base function and the spline, respectively. This formulation captures the core innovation of KANs, which aims to replace traditional weight-based linear transformations with coordinate-wise, learnable nonlinear functions. Each neuron in the KAN layer learns a unique nonlinear mapping that combines a smooth spline-based function with a residual base activation (SiLU) Z. Liu et al. (2025). This approach makes the network highly expressive and better suited for settings where input feature dimensions may vary between evaluation scenarios. Additionally, it enhances interpretability by allowing per-feature adaptive transformations.

Table 3.2: Extracted features and their type, equation, and explanation

Name	Type	Equation	Explanation
Mean	Statistical	$\bar{I} = \frac{1}{N}\sum_{i=1}^{N} I_i$	Mean is the average value of the current and voltage signals. I_i is the current at index i, and N is the total number of recordings.
Standard Deviation (std)	Statistical	$\sigma_I = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(I_i - \bar{I})^2}$	Measures the spread of current/voltage values around the mean. I_i is the ith current value, $\bar{I}$ is the mean, and N is the total count.
Skewness (skew)	Statistical	$S_I = \frac{1}{N}\sum_{i=1}^{N}\frac{(I_i-\bar{I})^3}{\sigma_I^3}$	Skewness describes the symmetry of the signal. I_i is the current value, $\bar{I}$ is the mean, σ_I is the standard deviation.
Peak	Statistical	$I_{\text{peak}} = \max(\lvert I_i \rvert)$	Maximum absolute value of the current or voltage signal.
Root Mean Square (RMS)	Statistical	$I_{\text{rms}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} I_i^2}$	Effective value of the varying current signal. I_i is the ith value; N is the total number of values.
Crest Factor (CF)	Statistical	$CF_I = \frac{I_{\text{peak}}}{I_{\text{rms}}}$	Indicates sharpness of peaks in the signal. CF is high for spiky waveforms.
Form Factor (FF)	Statistical	$FF_I = \frac{I_{\text{rms}}}{\bar{I}}$	Describes how peaky or flat the voltage and current signal is, where $\bar{I}$ and I_{rms} are the mean and RMS values of the current.
Kurtosis	Statistical	$K_I = \frac{1}{N}\sum_{i=1}^{N}\frac{(I_i-\bar{I})^4}{\sigma_I^4}$	Measures the tailedness of the waveform. I_i is the ith current sample, N is the total number of recordings, $\bar{I}$ is the mean, and σ_I is the standard deviation.
Apparent Power (S)	Power	$S = V_{\text{rms}} \times I_{\text{rms}}$	Apparent power represents the total power flowing in the circuit, including both active and reactive parts. Measured in volt-amperes (VA).
Active Power (P)	Power	$P = \frac{1}{N}\sum_{i=1}^{N} V_i I_i$	Actual power consumed by the appliance. V_i and I_i are voltage and current at time i.
Reactive Power (Q)	Power	$Q = \sqrt{S^2 - P^2}$	Power that sustains magnetic/electric fields in reactive components. Measured in VAR. S and P are apparent and active power, respectively.
Peak Power	Power	$P_{\text{peak}} = \max(P_i)$ $Q_{\text{peak}} = \max(Q_i)$ $S_{\text{peak}} = \max(S_i)$	Maximum values of active, reactive, and apparent power.
Minimum Power	Power	$P_{\text{min}} = \min(P_i)$ $Q_{\text{min}} = \min(Q_i)$ $S_{\text{min}} = \min(S_i)$	Minimum values of active, reactive, and apparent power.
Current Harmonics	Frequency Domain Features	$Har_{\text{I}} = \text{FFT}(I_i)$	First 25 harmonic components extracted using Fast Fourier Transform. I_i is the current waveform.

3.2.2 Feature Extraction

A total of 75 features from three different classes (or types) have been extracted from the raw current and voltage signals. These features are detailed in Table 3.2. All features presented in Table 3.2 were extracted for both current and voltage waveforms. Although 25 voltage harmonic features were initially computed, they demonstrated limited effectiveness in distinguishing appliance signatures. Consequently, these features were excluded from the final analysis, and the remaining 50 features were utilized for appliance classification.

3.3 Experiments

The experiments aim to evaluate the performance of KAN across the selected datasets and appliances. First, we describe our selected datasets. The experimental setup and training are then outlined. Finally, the results obtained from all tests are discussed in detail, and the best-performing ones are validated using 5-fold cross-validation, where the train-test splits are stratified by appliance type.

3.3.1 Data sets

Three Public datasets are selected to evaluate KAN's performance. The first dataset is the Controlled On/Off Loads Library (COOLL) dataset Picon et al. (2016). It contains current and voltage measurements for 12 different types of appliances, sampled at a rate of 100 kHz. These appliances are located in the PRISME Laboratory at the University of Orléans, France. The dataset includes the following appliances: drill, fan, grinder, hair dryer, hedge trimmer, lamp, paint stripper, planer, router, sander, saw, and vacuum cleaner. The second dataset is the Plug-Load Appliance Identification Dataset (PLAID) Medico et al. (2020). It has two versions namely PLAID1 and PLAID2. We are using PLAID1, which has current and voltage measurements of 11 different types of appliances sampled at 30 kHz. These appliances are present in 56 different houses in Pittsburgh, Pennsylvania, USA. The dataset includes the following appliances: air conditioner, bulb, compact fluorescent lamp, fan, fridge, hairdryer, heater, laptop, microwave, vacuum, and washing machine. The second dataset, A Worldwide Household and Industry Transient Energy Dataset (WHITED)

Kahl et al. (2016), contains current and voltage recordings for 110 different appliances, which can be grouped into 47 distinct types. The dataset includes data from households located in four regions of Germany, one in Austria, and two in Indonesia. The sampling rate of the data is 44 kHz.

Table 3.3: Characteristics of the datasets

Dataset	Collection Date	Buildings	Total Appliances	Types of Appliances	Variety	Characteristics	Sampling Frequency
WHITED	2015–2016	–	110	47	1–9	V, I	44 kHz
PLAID	Summer 2013	56	200	11	∼20	V, I	30 kHz
COOLL	June 2016	1	42	12	20–160	V, I	100 kHz

The characteristics of the datasets used in this study are presented in Table 3.3. It gives the data collection date, the number of buildings, the number of appliances, the number of appliance types (classes), and the number of appliances for each class (Variety). The reason for choosing the COOLL dataset is its high sampling frequency, unique set of appliances, and substantial number of samples for each appliance, in contrast to the other two datasets, which contain more appliance types but fewer samples per appliance. The main reason for choosing the PLAID dataset is that it contains recordings of different operating modes of appliances, which separates it from traditional load identification datasets. This characteristic helps models learn various initiation patterns. Secondly, we selected the WHITED dataset due to its larger number of appliances of different types, which allows the model to be trained on a diverse range of appliance categories.

3.3.2 Experimental Setup

The experiments were conducted using the following parameters: depending on the dataset complexity, the hyperparameters Grid (G) and k in KAN were varied. For example, since the number of appliances in the PLAID and COOLL datasets is smaller than in the WHITED dataset, k was set to 2 for PLAID and COOLL, and 3 for WHITED. To prevent overfitting, the value of G was kept constant at 10 across all datasets. Cross-entropy loss was used to monitor both training and test losses. The Limited-memory Broyden–Fletcher–Goldfarb–Shanno (LBFGS) algorithm D. C. Liu and Nocedal (1989) was employed as the optimizer, with a learning rate of 1 (the default value for LBFGS). To address overfitting, a regularization parameter, lamb of 0.01, was added to the model.

The following evaluation metrics have been used to analyze the performance of the model. Accuracy is the percentage of correct predictions that a trained classification model makes. It is calculated as Accuracy $= \frac{\sum_{i=1}^{N} TP_i + TN_i}{\sum_{i=1}^{N} (TP_i + FP_i + FN_i + TN_i)}$. Macro precision is the average of the per-class precision values, computed as $\text{P}_{\text{macro}} = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FP_i}$. Macro recall is the average of the per-class recall values and is given by $\text{R}_{\text{macro}} = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i}$. Macro F1 score is the average of the F1 scores for each class, where each class-specific F1 is the harmonic mean of its precision and recall: $\text{F1}_{\text{macro}} = \frac{1}{N} \sum_{i=1}^{N} 2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}$. Where TP_i and FP_i are true positives and false positives, respectively, while TN_i and FN_i are true negatives and false negatives, respectively.

3.3.3 Training

The data was split into 80% for training and 20% for testing. 80% of the data was used for training and fed directly into the KAN after feature extraction. The hyperparameters were tuned to minimize the cross-entropy loss and improve accuracy. The learning rate remained constant throughout all experiments during training.

3.3.4 Results

This section provides a detailed analysis of all the tests and examines the results based on different combinations of features. The model was trained using various feature subsets to identify the most effective ones for achieving the highest accuracy. Furthermore, the model's performance is evaluated using evaluation metrics and various graphical representations. The performance of the proposed model has been compared against benchmark models De Baets, Ruyssinck, Develder, Dhaene, and Deschrijver (2018); De Baets et al. (2017); De Baets, Dhaene, Deschrijver, Develder, and Berges (2018); Dimbiniaina, Pau, and Naramo (2023); Le, Heo, and Kim (2021); Mulinari et al. (2019) to evaluate its relative effectiveness.

For each dataset, voltage harmonics were extracted but not utilized in any tests, as they were found to be strongly correlated with one another. Additionally, most voltage features, when used to train the model, resulted in decreased performance and increased computational cost. This is because the voltage across most appliances is quite similar, and for many Type 1, 2, and 3 appliances, the voltage stabilizes after initiation. For harmonic features, only the 1st to 6th and 20th to 25th

harmonics were used for training in all the datasets, since the remaining harmonics were highly correlated with each other, as illustrated in Figures 3.4 and 3.6. Tables 3.4, 3.5, and 3.7 present the results of the model on the COOLL, PLAID, and WHITED datasets, respectively. Where G and K are hyperparameters for KAN, q is reactive power, cf is crest factor, ff is form factor, std is standard deviation, and skew is skewness. The first column specifies the type of features used in each test, while the second column lists the selected features used to train the model. Columns 3 and 4 show the accuracy and F1 score, respectively.

Table 3.4: KAN's parameters, hyperparameters, and performance on different types and combinations of features for COOLL dataset. The best results are highlighted in bold.

Type	**Combinations**	**Accuracy (%)**	**F1 (%)**	**Network**
Individual	i_cf, v_cf	56	36	2, G10, K=2
Individual	i_RMS, v_RMS	40	21	2, G10, K=2
Individual	i_ff, v_ff	16	6	2, G10, K=2
Individual	i_peak, v_peak	35	12	2, G10, K=2
Individual	i_kurtosis, v_kurtosis	48	23	2, G10, K=2
Individual	i_std, v_std	19	3	2, G10, K=2
Individual	i_skew, v_skew	32	24	2, G10, K=2
Individual	i_mean, v_mean	19	3	2, G10, K=2
Selective i Harmonics	**i harmonics (1–6)**	**96**	**96**	**2, G10, K=2**
i Harmonics	All i Harmonics	95	94	2, G10, K=2
All Power	Power Features (RMS, Peak, Min)	91	92	2, G10, K=2
i Statistical	i_cf, i_ff, i_RMS, i_kurtosis, i_mean, i_peak, i_skew, i_std	76	72	2, G10, K=2
Power & i Harmonics	All Power Features (RMS, Peak, Min), i_harmonics (1–6, 19–25)	95	95	2, G10, K=2
All	All 50 features	9	3	2, G10, K=2

Table 3.4 presents the results of the KAN on the COOLL dataset. The first eight tests were conducted on individual combinations of current and voltage statistical features. Among them, the

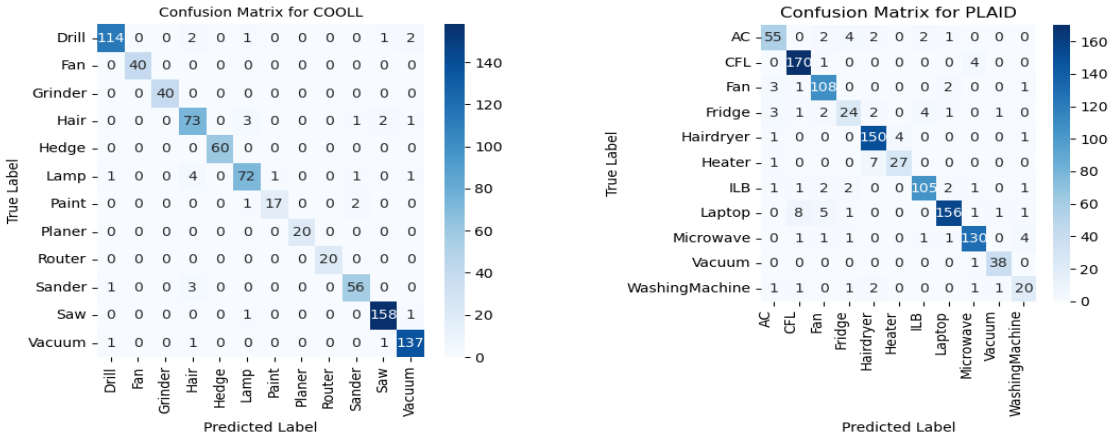

Figure 3.2: Confusion matrices for COOLL and PLAID best test from Tables 3.4 and 3.5. Hedge = hedge trimmer, Paint = paint stripper, Vacuum = vacuum cleaner, CFL = compact fluorescent lamp, ILB = incandescent light bulb, AC = air conditioner.

current and voltage Crest Factor were the most effective, achieving an accuracy of 56% and an F1 score of 36%. In contrast, as shown in Tests 3, 6, and 8, features such as Form Factor, standard deviation, and mean were the least effective. Current and voltage kurtosis showed performance close to the Crest Factor due to their high correlation. Compared to harmonics and power features, none of the statistical features achieved competitive performance. When all statistical features were combined in Test 12, the accuracy and F1 score reached 76% and 72%, respectively, still lower than the selective harmonic features.

In Test 10, using all harmonic features, the model achieved 95% accuracy and 94% F1. However, most harmonic features are highly correlated, and higher-order harmonics become smaller and less informative. To address this, only the first six harmonics were used in Test 9, yielding 96% accuracy and 96% F1. Meanwhile, Test 11 used only power features, achieving 91% accuracy and 92% F1. Although power features are highly effective, their strong correlation can lead the model to learn incorrect patterns. This is evident in Test 14, where using all features caused a drastic drop in performance down to 9% accuracy and 3% F1. Based on these findings, the final test utilized the most effective and uncorrelated features (power and harmonics), resulting in 95% accuracy and 95% F1 score, which is very close to results from the selective harmonics (1-6). These results are highlighted in bold in Table 3.4.

Figure 3.2 displays the confusion matrix corresponding to the best-performing test on the COOLL

Table 3.5: KAN's parameters, hyperparameters, and performance on different types and combinations of features for PLAID dataset. The best results are highlighted in bold.

Type	Combinations	Accuracy (%)	F1 (%)	Network
Individual	i_cf, v_cf	54	47	2, G10, K=2
Individual	i_RMS, v_RMS	11	2	2, G10, K=2
Individual	i_ff, v_ff	34	16	2, G10, K=2
Individual	i_peak, v_peak	56	43	2, G10, K=2
Individual	i_kurtosis, v_kurtosis	53	37	2, G10, K=2
Individual	i_std, v_std	55	37	2, G10, K=2
Individual	i_skew, v_skew	49	31	2, G10, K=2
Individual	i_mean, v_mean	33	27	2, G10, K=2
Selective i Harmonics	i harmonics (1–6)	72	62	2, G10, K=2
i Harmonics	All i Harmonics	78	69	2, G10, K=2
All Power	**Power Features (RMS, Peak, Min)**	**92**	**88**	**2, G10, K=2**
i Statistical	i (cf, ff, kurtosis, mean, skew, std, peak, RMS)	72	49	2, G10, K=2
Power & i Harmonics	All Power Features (RMS, Peak, Min), i_harmonics (1–6)	87	83	2, G10, K=2
All	All 50 features	73	48	2, G10, K=2
Selective (Power & i Harmonics & Statistical)	i harmonics (2–6), q – i (RMS), i_mean, i_peak, q_min	85	78	2, G10, K=2

dataset, as reported in Table 3.4. Certain appliances such as the fan, grinder, planer, router, and hedge trimmer achieved perfect classification, with no instances misclassified. Among all appliances, the paint stripper had the highest number of misclassifications relative to its total sample count. Despite this, the majority of appliances in the COOLL dataset were classified correctly, leading to strong overall performance KAN achieved 96% accuracy and F1-score, while Random Forest attained 99% accuracy and F1-score.

Table 3.5 presents the results on the PLAID dataset. Initially, similar to the COOLL dataset, the model was trained and tested on individual statistical feature combinations. Among them, the Crest Factor (cf) showed the highest performance, achieving an accuracy of 54% and an F1 score of 47%. Tests using current and voltage kurtosis produced comparable results, which is expected due to their strong correlation with the Crest Factor (as shown in Figure 3.3). Therefore, only one of these features was selected for the last test. Moreover, features such as form factor (ff) and the

mean of current and voltage produced approximately the same accuracy and F1 scores. Therefore, only one feature from this group was selected for the last test to avoid unnecessary computational overhead. A similar approach was applied to other features, such as skewness, standard deviation (std), and peak values of current and voltage; only one feature was selected.

In Test 10, the model was trained using all 25 harmonics, achieving an accuracy of 78% and an F1 score of 69%, outperforming all tests using statistical features. However, Figure 3.4 shows that all harmonics beyond the 6th are strongly correlated with each other. Including all 25 harmonics in training, as in Test 9, increased computational complexity while yielding only a marginal performance improvement. By selecting only the uncorrelated harmonics (1st–6th) for training, we achieved 72% accuracy and an F1 score of 62%, which is 6% lower than the result using all harmonics.

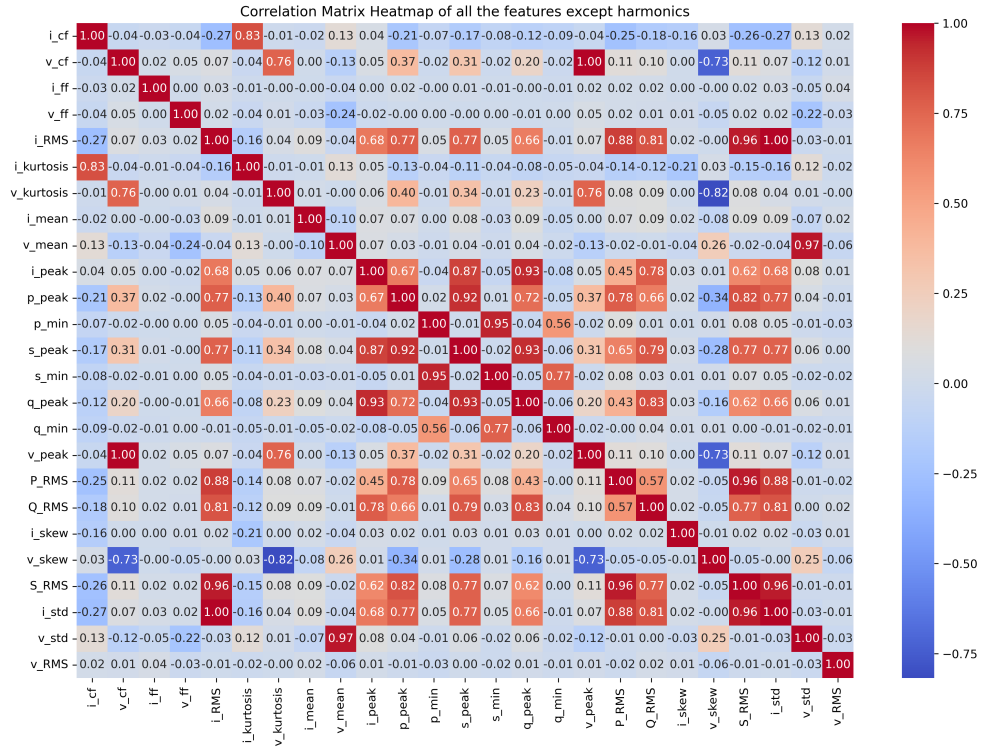

Figure 3.3: The correlation matrix of features (excluding harmonic features) extracted from the PLAID dataset appliances shows several strong correlations.

In the 11th test, we used all the power-related features, which resulted in an accuracy of 92% and an F1 score of 88%. Combining selective harmonic features with power features yielded slightly

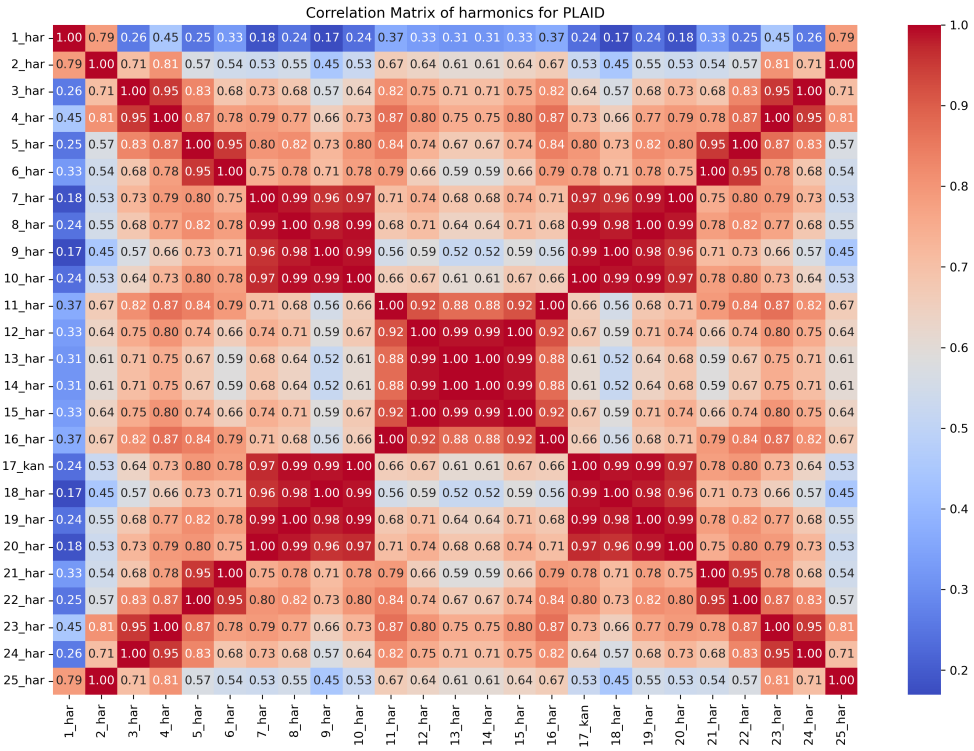

Figure 3.4: The correlation matrix for current harmonics extracted from the PLAID dataset shows that, except for the 1st to 6th harmonics, all remaining harmonics are strongly correlated with each other.

lower performance (as shown in Test 11 of Table 3.5). Using all available features simultaneously led to a decline in model performance, as the inclusion of many non-contributing or less informative features distracted the model, causing it to learn incorrect patterns and ultimately reducing accuracy. In the final test, we selected only the most informative and uncorrelated features. Due to the strong correlation between current RMS and both active and apparent power, only current RMS was retained. Similarly, among the minimum power values, only one was selected, and all power peak features were excluded due to their correlation with current peak values. This refined feature selection led to strong model performance, achieving 85% accuracy and an F1 score of 78%. (Note: Although strong correlations between harmonics and other features were observed, the corresponding matrix is not included in this paper due to its large size; it is available on GitHub.)

Compared to the COOLL dataset, the model's performance decreased when trained on the most uncorrelated features in this case. This decline can be attributed to the increased number and diversity of appliances in the dataset. As the number of appliance types increases, power features become

more effective and informative for distinguishing between appliances.

Table 3.6: Appliance-wise comparison for PLAID. The first 3 models (Logistic Reg, RF, Neural Net) are from De Baets et al. (2017). RF = Random Forest, CFL = compact fluorescent lamp, AC = air conditioner, ILB = incandescent light bulb, C = Correct predictions, I = Incorrect predictions

Appliance	**Logistic Reg**		**RF**		**Neural Net**		**KAN**		**RF**	
	C	I	C	I	C	I	C	I	C	I
Heater	1	34	0	35	0	35	27	8	20	8
Washing Machine	14	12	15	11	14	12	20	6	14	7
Laptop	158	14	155	17	162	10	156	16	137	1
CFL	160	15	163	12	161	14	170	5	137	3
Microwave	116	23	130	9	129	10	130	9	109	2
Fridge	7	31	14	24	12	26	24	14	13	17
Fan	43	72	62	53	60	55	108	7	90	2
Vacuum	36	2	31	7	38	0	38	0	30	0
AC	14	52	21	45	19	47	55	11	43	10
Hairdryer	141	15	137	19	143	13	150	6	121	4
ILB	97	17	103	11	108	6	105	9	85	6
Accuracy (%)	73		77		79		92		**93**	

For PLAID dataset, we have conducted a comparative appliance-wise evaluation with De Baets et al. (2017). A valuable feature for appliance classification is the voltage-current (V-I) trajectory. In this study, the V-I trajectory is transformed into a binary image, and the contours within these images are extracted. From these contours, elliptic Fourier descriptors are computed and used as input features for classification models De Baets et al. (2017). To evaluate the effectiveness of features, experiments were conducted using Logistic Regression, Random Forest, and a Neural Network that shares the same structure as the KAN, but with a different number of neurons.

Figure 3.2 shows the confusion matrix for the best performing test on the PLAID dataset, while Table 3.6 summarizes the results from each classification model, including those from KAN and Random Forest tested on the features extracted in this study. For the heater, all three baseline models from De Baets et al. (2017) show difficulty in classification, frequently misidentifying it as a hairdryer. This confusion arises because both devices use heating coils. However, KAN reduces this error, with only 8 instances of heater misclassified as hairdryer. In the case of the washing machine, performance has also improved, with only 6 misclassifications. This improvement can be

attributed to PLAID's inclusion of samples from multiple operation cycles, except for the startup phase. For appliances such as microwave, compact fluorescent lamp (CFL), vacuum, incandescent light bulb (ILB), hairdryer, and laptop, all models demonstrate similar classification behavior. A small number of laptop instances are misclassified as CFL, likely due to the laptop's low power consumption. A significant improvement is observed in classifying the fridge. While traditional machine learning models struggled with this appliance, KAN successfully classifies twice as many instances correctly. Nonetheless, the fridge remains the most frequently misclassified appliance overall, which highlights the complexity of its power signature.

Table 3.7: KAN's parameters, hyperparameters, and performance on different types and combinations of features for WHITED dataset. The best results are highlighted in bold.

Type	**Combinations**	**Accuracy (%)**	**F1 (%)**	**Network**
Individual	i_cf, v_cf	20	3	2, G10, K=3
Individual	i_RMS, v_RMS	21	4	2, G10, K=3
Individual	i_ff, v_ff	10	2	2, G10, K=3
Individual	i_peak, v_peak	16	2	2, G10, K=3
Individual	i_kurtosis, v_kurtosis	20	13	2, G10, K=3
Individual	i_std, v_std	18	3	2, G10, K=3
Individual	i_skew, v_skew	24	6	2, G10, K=3
Individual	i_mean, v_mean	7	1	2, G10, K=3
Selective i Harmonics	i harmonics (1–6)	53	44	2, G10, K=3
i Harmonics	All i Harmonics	61	55	2, G10, K=3
All Power	**Power Features (RMS, Peak, Min)**	**85**	**85**	**2, G10, K=3**
i Statistical	i (cf, ff, kurtosis, mean, skew, std, peak, RMS)	68	53	2, G10, K=3
Power & i Harmonics	All Power Features (RMS, Peak, Min), i_harmonics (1–6)	78	69	2, G10, K=3
All	All 50 features	68	53	2, G10, K=3

Table 3.7 presents the results of the KAN on different feature combinations from WHITED dataset. It follows the same structure as Table 3.5. The same experimental strategy used for the PLAID dataset was applied to the WHITED dataset. Similar combinations of features were selected to evaluate the model's performance and identify the most contributing features.

The main difference between the PLAID and WHITED datasets lies in the diversity of appliance

types; WHITED includes over three times more appliance types than PLAID, but with limited instances per type. Due to this increase in data complexity caused by a larger number of appliance types with fewer examples each, the parameter K was set to 3 for all tests on the WHITED dataset. K refers to the number of internal grid points used in the learnable function of each neuron in the KAN Z. Liu et al. (2025). When the complexity of the problem increases, a higher K is required to capture the complex patterns. However, a larger K increases computational cost and the risk of overfitting.

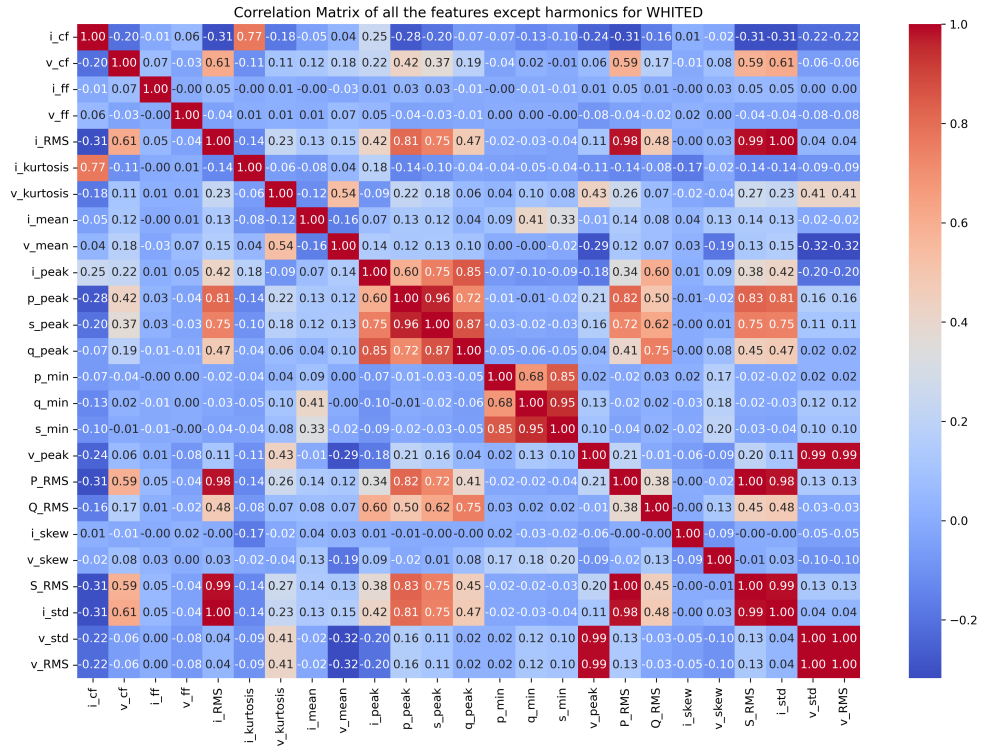

Figure 3.5: The correlation matrix of features (excluding harmonic features) extracted from the WHITED dataset appliances shows that most of the features are strongly uncorrelated to each other.

Due to the diversity and limited number of instances in WHITED, we ran multiple tests on each feature combination with different hyperparameters and reported the average accuracy and F1 score in Table 3.7. The limited data per appliance type makes it difficult for the model to learn meaningful patterns. Increasing Grid and K values to improve learning under such diversity often leads to overfitting.

Initially, we tested different combinations of statistical features. No combination of statistical

features achieved more than 24% accuracy or 13% F1 score. The highest accuracy (24%) was obtained using skewness values of current and voltage, while the highest F1 score (13%) came from using the kurtosis values of current and voltage. All other statistical combinations showed similar performance. The average accuracy across 8 statistical feature combinations was 17%, and the average F1 score was 4.25%. Using all statistical current features slightly improved the results, approximately 51% higher than the average accuracy and F1 score of combined current and voltage statistical features. The underperformance of statistical features on the WHITED dataset can be attributed to several factors. First, statistical features typically require a substantial number of signal samples with sufficient temporal variation to capture the signal's characteristics reliably. However, the WHITED dataset contains a limited number of samples per appliance, which results in statistical features that are unstable and unrepresentative. Furthermore, the dataset comprises appliances from 47 distinct classes, with each class having only 1 to 9 samples (as shown in Table 3). This imbalance increases the likelihood of feature overlap, where multiple appliances exhibit similar or identical values for features such as mean, standard deviation, or RMS, thereby diminishing the discriminative power of these features.

Harmonics in the WHITED dataset were found to be slightly less correlated compared to those in PLAID (see Figure 3.6). An initial test using only the first 6 harmonics yielded 53% accuracy and 44% F1, which is higher than the statistical feature combinations. When all harmonics were used, the model achieved 61% accuracy and 55% F1, highlighting the importance of harmonic features in handling diverse data. Testing the model with all power features resulted in 85% accuracy and 84% F1. Based on these results, we combined both harmonic and power features for training, achieving the average performance of 78% accuracy and 69% F1 (averaged across models with varying parameters). The last 6 harmonics were also included since their correlation is approximately the same as the first 6 harmonics (as shown in Figure 3.6). A final test using all available features resulted in only 68% accuracy and 53% F1.

Figure 3.7 presents the confusion matrix for the best test case on the WHITED dataset from Table 3.7. Similar to the patterns observed in the PLAID and COOLL datasets, hairdryer remains a commonly misclassified appliance. In WHITED, appliances such as hairdryer, iron, and kettle are often confused with one another due to their use of heating coils. For the same reason, stove is

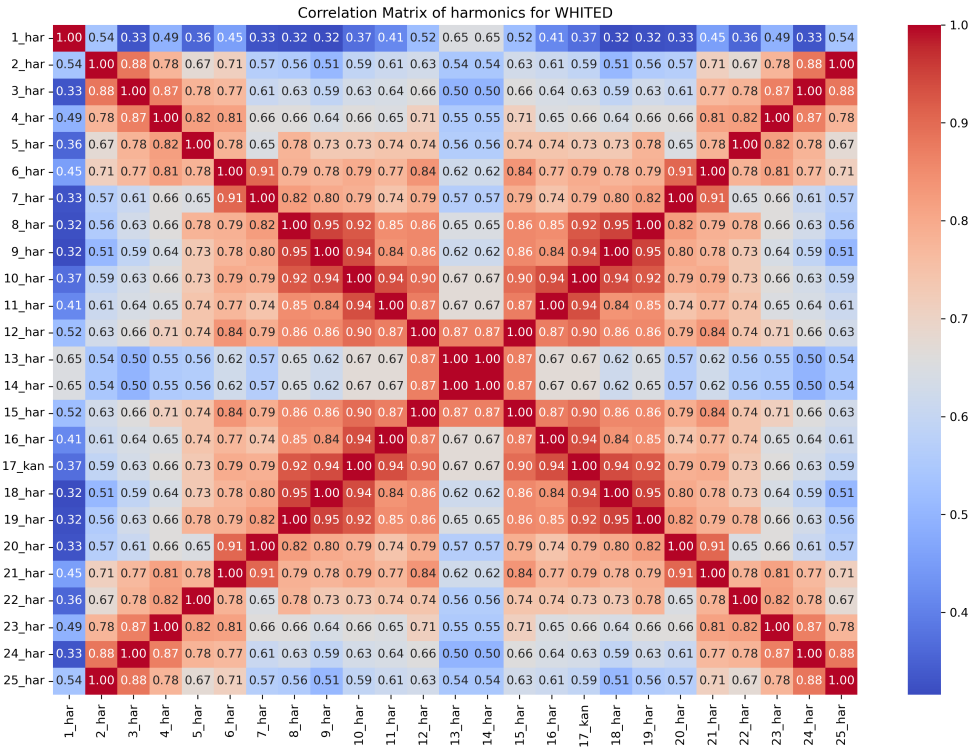

Figure 3.6: The correlation matrix for current harmonics extracted from the WHITED dataset.

frequently misclassified as a kettle, and the kettle is also confused with the toaster and water heater. Additionally, low power-consuming appliances like LEDs, lightbulbs, and power supplies are misclassified as other devices with similar consumption. Lastly, the water heater is often confused for a fan heater, likely due to their similar operational mechanisms. Aside from these cases, most other appliances in the WHITED dataset show minimal misclassification.

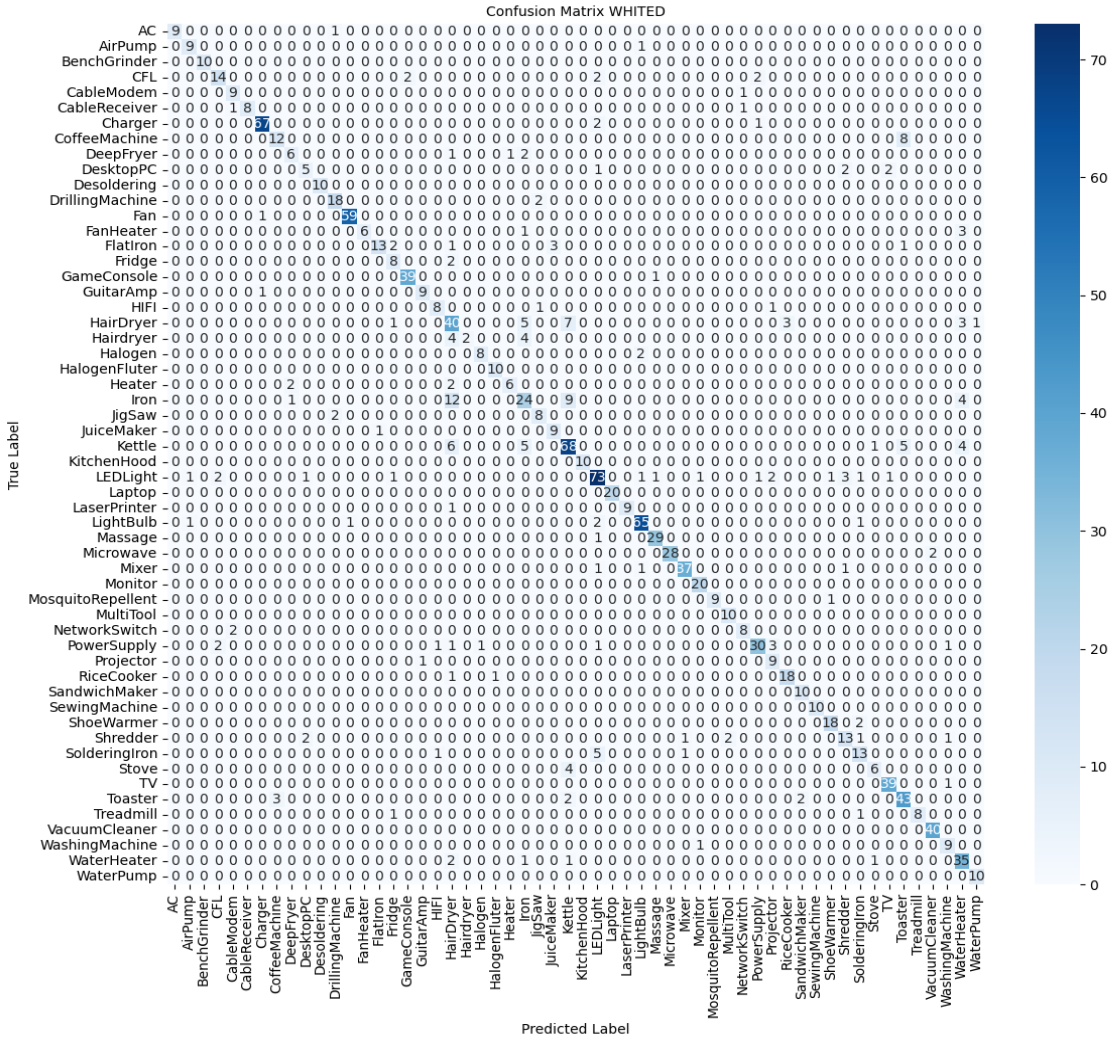

Figure 3.7: Confusion matrix of WHITED dataset for the best test from Table 3.7

Table 3.8 presents the comparative results of different benchmark approaches along with KAN and Random Forest. In De Baets et al. (2018), a weighted pixel-based image representation of the V–I trajectory is employed as input to a convolutional neural network (CNN). The model was tested on PLAID and WHITED datasets, resulting in F1 of 78% and 75% respectively. Our proposed models, KAN and Random Forest, demonstrated improved performance in terms of F1 score. In Le et al. (2021), the authors introduce a novel approach called HT-LSTM (Hilbert Transform Long Short-Term Memory). The proposed method has two main components: (i) it extracts a new transient feature using the Hilbert Transform, referred to as APF (Amplitude-Phase-Frequency), which

captures sequential information; and (ii) it utilizes a Sequence-to-Sequence Long Short-Term Memory (Seq2Seq LSTM) model to classify appliances based on the extracted APF features. The model was evaluated on PLAID dataset, achieving an F1 of 95% and accuracy of 90%. In contrast, our proposed methods KAN and Random Forest outperform in terms of accuracy. They showed relatively lower F1 scores, indicating room for improvement in class-wise prediction consistency.

Table 3.8: Comparison of different models across datasets with accuracy, F1-score, and number of classes.

Model	**Dataset**	**Accuracy (%)**	**F1-score (%)**	**Number of classes**
CNN	PLAID	–	78	All
De Baets et al. (2018)	WHITED	–	75	All
HT-LSTM	PLAID	90	95	All
Le et al. (2021)				
Neural Net	PLAID	77	–	6
Dimbiniaina et al. (2023)	COOLL	96	–	5
	WHITED	96	–	7
CNN	PLAID	–	88	All
De Baets et al. (2018)				
kNN	COOLL	96	–	All
Mulinari et al. (2019)				
Ensemble	COOLL	99	–	All
Mulinari et al. (2019)				
SVM	COOLL	96	–	All
Mulinari et al. (2019)				
KAN	PLAID	92	88	All
	WHITED	85	85	All
	COOLL	96	96	All
RF	PLAID	93	93	All
	WHITED	96	96	All
	COOLL	99.40	99.40	All

In Dimbiniaina et al. (2023), Mel power spectrogram is for the feature extraction from voltage and current signals. This study explores the possibility of approximating the Mel power spectrogram using compact neural networks. The proposed method enables the construction of an end-to-end, multitask deep learning pipeline. The approach was evaluated on COOLL, PLAID, and WHITED datasets, resulting in the accuracy of 77%, 96%, and 96% respectively, on each dataset. But only 6 classes from PLAID, 5 classes from COOLL, and 7 classes from WHITED have been selected

for the tests, which justifies the high accuracy of the proposed approach. In all the tests for our proposed models, all the classes (shown in Table 3.3) have been selected, and the accuracy is still very close to the benchmark model's accuracy. In De Baets et al. (2018), the authors proposed 2 models. A random forest with elliptical Fourier descriptors for VI trajectories of appliances and a CNN with VI images of appliance signals. The models were evaluated on PLAID, achieving the highest F1 of 88. In contrast, KAN results in the same F1; meanwhile, Random Forest outperforms by 93% of F1.

In Mulinari et al. (2019), the authors proposed an extended feature extraction technique based on VI trajectories, introducing new steady-state and transient features. They evaluated their approach using three classifiers: K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and an ensemble method on the COOLL dataset, achieving accuracies of 96%, 96%, and 99% respectively. Our proposed KAN model achieved comparable accuracy to KNN and SVM, while slightly underperforming compared to the ensemble. On the other hand, the Random Forest classifier surpassed both KNN and SVM, matching the ensemble's top performance with an accuracy of 99%.

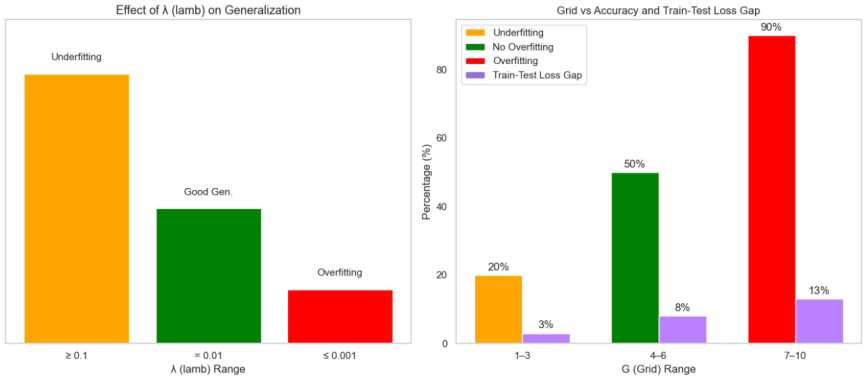

Figure 3.8: Effect of KAN's hyperparameters (G, lamb) on overfitting and generalization.

Figure 3.8 shows the effect of different values of hyperparameters Grid (G) and lamb on generalization and overfitting in KAN. On the left side, the bar chart illustrates the impact of different values of the regularization parameter λ on overfitting. The x-axis represents the values of λ. On the right side, the bar chart highlights two key aspects: (i) how the grid size parameter G influences overfitting, and (ii) the gap between training and testing loss. The x-axis indicates the values of G, and the y-axis represents the model accuracy. When G is set between 1 and 3, KAN exhibits

underfitting, with an accuracy of approximately 20% and a train-test loss gap of about 3%. As G increases to values between 7 and 10, the model begins to overfit, resulting in a widened train-test loss gap of nearly 13%, achieving an accuracy of around 90%. For intermediate G values, the model demonstrates better generalization, with an accuracy of roughly 50%. However, when a regularization term of $\lambda = 0.01$ is applied alongside higher G values (7–10), overfitting is substantially suppressed, and the train-test loss gap is reduced back to approximately 3%, while maintaining a high accuracy of 90%. While this does not eliminate overfitting completely, it helps suppress it to a significant extent.

Chapter 4

Conclusion

This thesis presents machine learning models for NILM and appliance identification, incorporating various techniques for feature selection, amplification, and engineering. The models are evaluated using different evaluation metrics across multiple datasets collected from different geographical regions.

In chapter 2, an FFNN model is enhanced through oversampling and feature amplification to improve energy disaggregation performance. For better pattern recognition, we employed oversampling for training data. Most energy datasets contain two main power lines, with some appliances connected to only one line and others to both. Instead of combining them, both aggregated power features have been used for training the model. Experiments were conducted on the RAE, REDD, and REFIT datasets, assessing both noisy and noise-free scenarios for the REFIT data. For simplicity, only a single household was selected for evaluation. In the noisy REFIT scenario, model performance deteriorates due to combined aggregate profiles and a high number of activation cycles per appliance, which requires undersampling. For the RAE and REDD datasets, model performance is benchmarked against the DAE model, proposed in Bonfigli et al. (2018). For the RAE and REDD datasets, based on F1 and NEP, the FFNN model outperforms the DAE model in general performance. The primary objective of this model is to provide a computationally efficient solution for energy disaggregation. Lastly, an appliance-wise comparison has been conducted for washers on the datasets from the three different regions. It shows that appliances from different regions, non-uniform patterns in the appliance activations, aggregated mains, and a small number of activation

cycles can affect the models performance negatively. The appliance-wise comparison has been limited to the washer across three datasets from three distinct regions. The idea of using lighter models combined with techniques like oversampling and a greater number of aggregate features proves that the energy disaggregation performance can be improved even with limited computational resources.

In chapter 3, Kolmogorov–Arnold Networks (KAN) and Random Forest models have been employed to identify appliances. Both models were evaluated using 75 extracted features across three datasets COOLL, PLAID, and WHITED. KAN was tested with various feature combinations. By using correlation matrices and conducting multiple experiments, the most informative and uncorrelated features were identified for effective appliance classification. Furthermore, the best-performing experiments were validated using 5-fold cross-validation, with train-test splits stratified by appliance type. Compared to the WHITED and PLAID dataset, the KAN performed better on the COOLL dataset. This demonstrates that as the number of appliance types in a dataset increases, making it more diverse, the model's performance tends to decline. Another contributing factor to this decline is that although the WHITED and PLAID datasets include more appliance types and appliances, its overall size remains approximately the same as COOLL dataset. This results in fewer samples per appliance type, providing insufficient data for the model to learn each pattern effectively. The model's behavior varied significantly across the two datasets. Additionally, statistical features played an important role in appliance identification for the COOLL dataset but were ineffective for the PLAID and WHITED datasets due to the high degree of uncorrelated features. As a result, these features contributed little to performance in more diverse datasets. In contrast, harmonic and power features were valuable for all the datasets. Lastly, to overcome the overfitting in KAN, hyperparameter tuning was introduced. KAN is a highly efficient model, especially for simpler problems. For complex problems, the model requires higher K and Grid values to learn intricate patterns. As noted earlier, increasing these values often leads to overfitting. To reduce overfitting, regularization has been used. This challenge can also be addressed in two ways: (1) increasing the amount of labeled data, and (2) increasing the number of hidden layers while decreasing K and Grid, which significantly raises computational requirements. By using KAN and Random Forest shows the need for advancements in the deep learning techniques as compared to

traditional machine learning models. Furthermore, as the number and variety of appliances is continuously increasing in modern households, supervised models need more labeled or augmented data for each appliance for better identification.

Future research can address the challenge of data scarcity by employing data augmentation techniques based on existing labeled data. The Incorporation of additional features could further enhance disaggregation accuracy, such as the average activation duration and the use of dual aggregate profiles, considering that many regions supply power through two main lines per building. The integration of more informative features for appliance recognition can also be beneficial. These features may include Total Harmonic Distortion (THD), Zero Crossing Rate (ZCR), Autocorrelation Function (ACF), Power Factor (PF), Fundamental Frequency (f_1), Dominant Harmonic Component, Spectral Entropy, Bandwidth (BW), and waveform Slope. These features may enhance the discriminative capacity of models and should be considered in future investigations. Additionally, exploring new categories of appliances for disaggregation could further broaden the scope and effectiveness of NILM systems. Lastly, in regards to the KAN architecture, it is recommended to explore methods to mitigate the overfitting issue and improve its generalizability.

References

Abubakar, I., Khalid, S., Mustafa, M., Shareef, H., & Mustapha, M. (2017). Application of load monitoring in appliances' energy management – a review. *Renewable and Sustainable Energy Reviews*, *67*, 235-245. doi: https://doi.org/10.1016/j.rser.2016.09.064

Akbar, M. K., Amayri, M., Bouguila, N., Delinchant, B., & Wurtz, F. (2024). Evaluation of regression models and bayes-ensemble regressor technique for non-intrusive load monitoring. *Sustainable Energy, Grids and Networks*, *38*, 101294. doi: https://doi.org/10.1016/j.segan.2024.101294

Altrabalsi, H., Liao, J., Stankovic, L., & Stankovic, V. (2014). A low-complexity energy disaggregation method: Performance and robustness. In *2014 ieee symposium on computational intelligence applications in smart grid (ciasg)* (p. 1-8). doi: 10.1109/CIASG.2014.7011569

Angelis, G.-F., Timplalexis, C., Krinidis, S., Ioannidis, D., & Tzovaras, D. (2022, 02). Nilm applications: Literature review of learning approaches, recent developments and challenges. *Energy and Buildings*, *261*, 111951. doi: 10.1016/j.enbuild.2022.111951

Athanasiadis, C., Doukas, D., Papadopoulos, T., & Chrysopoulos, A. (2021, 02). A scalable real-time non-intrusive load monitoring system for the estimation of household appliance power consumption. *Energies*, *14*, 747. doi: 10.3390/en14030767

Bonfigli, R., Felicetti, A., Principi, E., Fagiani, M., Squartini, S., & Piazza, F. (2018). Denoising autoencoders for non-intrusive load monitoring: Improvements and comparative evaluation. *Energy and Buildings*, *158*, 1461-1474. doi: https://doi.org/10.1016/j.enbuild.2017.11.054

Bucci, G., Ciancetta, F., Fiorucci, E., & Mari, S. (2020). Load identification system for residential applications based on the nilm technique. In *2020 ieee international instrumentation*

and measurement technology conference (i2mtc) (p. 1-6). doi: 10.1109/I2MTC43012.2020.9128599

Chouchene, S., Amayri, M., & Bouguila, N. (2024). Sparse coding-based transfer learning for energy disaggregation. *Energy and Buildings*, *320*, 114498. doi: https://doi.org/10.1016/j.enbuild.2024.114498

De Baets, L., Ruyssinck, J., Develder, C., Dhaene, T., & Deschrijver, D. (2018). Appliance classification using vi trajectories and convolutional neural networks. *Energy and Buildings*, *158*, 32-36. doi: https://doi.org/10.1016/j.enbuild.2017.09.087

de Aguiar, E. L., Bernardi, R., Lazzaretti, A. E., Pipa, D. R., Carati, E. G., & Cardoso, R. (2025). Load identification with photovoltaic distributed generation and a novel public high-frequency dataset. *Journal of Control, Automation and Electrical Systems*, *36*(2), 345–356.

De Baets, L., Develder, C., Dhaene, T., & Deschrijver, D. (2017). Automated classification of appliances using elliptical fourier descriptors. In *2017 ieee international conference on smart grid communications (smartgridcomm)* (p. 153-158). doi: 10.1109/SmartGridComm.2017.8340669

De Baets, L., Dhaene, T., Deschrijver, D., Develder, C., & Berges, M. (2018). Vi-based appliance classification using aggregated power consumption data. In *2018 ieee international conference on smart computing (smartcomp)* (p. 179-186). doi: 10.1109/SMARTCOMP.2018.00089

Dimbiniaina, M., Pau, D. P., & Naramo, T. A. (2023). Mel power spectrogram approximation by tiny neural networks for home appliances classification. In *2023 ieee international workshop on metrology for industry 4.0 & iot (metroind4.0&iot)* (p. 60-65). doi: 10.1109/MetroInd4.0IoT57462.2023.10180197

Faustine, A., Mvungi, N. H., Kaijage, S. F., & Kisangiri, M. (2017). A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem. *CoRR*, *abs/1703.00785*.

Figueiredo, M., de Almeida, A., & Ribeiro, B. (2012). Home electrical signal disaggregation for non-intrusive load monitoring (nilm) systems. *Neurocomputing*, *96*, 66-73. doi: https://doi.org/10.1016/j.neucom.2011.10.037

Gao, Y., Zhang, J., Wang, M., Tan, Z., & Liang, M. (2025). A lightweight load identification model update method based on channel attention. *Energies*, *18*(11). doi: 10.3390/en18112885

Gillis, J. M., Alshareef, S. M., & Morsi, W. G. (2016). Nonintrusive load monitoring using wavelet design and machine learning. *IEEE Transactions on Smart Grid*, *7*(1), 320-328. doi: 10.1109/TSG.2015.2428706

Hart, G. (1992). Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, *80*(12), 1870-1891. doi: 10.1109/5.192069

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. doi: 10.1109/MCSE.2007.55

Jiang, J., Wang, Z., Qiu, S., Li, X., & Zhang, C. (2025). Multi-task load identification and signal denoising via hierarchical knowledge distillation. *IEEE Transactions on Network Science and Engineering*, *12*(3), 1967-1980. doi: 10.1109/TNSE.2025.3542409

Kahl, M., Haq, A. U., Kriechbaumer, T., & Jacobsen, H.-A. (2016, 05). Whited-a worldwide household and industry transient energy data set. In *3rd international workshop on non-intrusive load monitoring* (pp. 1–4).

Kalinke, F., Bielski, P., Singh, S., Fouché, E., & Böhm, K. (2021). An evaluation of nilm approaches on industrial energy-consumption data. In *Proceedings of the twelfth acm international conference on future energy systems* (p. 239–243). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3447555.3464863

Kelly, J., & Knottenbelt, W. (2015). 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* (p. 55–64). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/2821650.2821672

Kolter, J., & Johnson, M. (2011, 01). Redd: A public data set for energy disaggregation research. *Artif. Intell.*, *25*.

Kotsilitis, S., Marcoulaki, E. C., & Kalligeros, E. (2024). A versatile, low-cost monitoring device suitable for non-intrusive load monitoring research purposes. *Measurement: Sensors*, *32*, 101081. doi: https://doi.org/10.1016/j.measen.2024.101081

Le, T.-T.-H., Heo, S., & Kim, H. (2021). Toward load identification based on the hilbert transform

and sequence to sequence long short-term memory. *IEEE Transactions on Smart Grid*, *12*(4), 3252-3264. doi: 10.1109/TSG.2021.3066570

Liang, H., & Ma, J. (2020). Separation of residential space cooling usage from smart meter data. *IEEE Transactions on Smart Grid*, *11*(4), 3107-3118. doi: 10.1109/TSG.2020.2965958

Linh, N. V., & Arboleya, P. (2019). Deep learning application to non-intrusive load monitoring. In *2019 ieee milan powertech* (p. 1-5). doi: 10.1109/PTC.2019.8810435

Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Math. Program.*, *45*(1-3), 503–528. doi: 10.1007/BF01589116

Liu, Y., Liu, W., Shen, Y., Zhao, X., & Gao, S. (2021). Toward smart energy user: Real time non-intrusive load monitoring with simultaneous switching operations. *Applied Energy*, *287*, 116616. doi: https://doi.org/10.1016/j.apenergy.2021.116616

Liu, Y., Wang, Y., Hong, Y., Shi, Q., Gao, S., & Huang, X. (2021). Toward robust non-intrusive load monitoring via probability model framed ensemble method. *Sensors*, *21*(21). doi: 10.3390/s21217272

Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljacic, M., ... Tegmark, M. (2025). KAN: Kolmogorov–arnold networks. In *The thirteenth international conference on learning representations.*

Lu, X., Chen, D., Geng, L., Wang, Y., Sheng, D., & Chen, R. (2025). Research on a single-load identification method based on color coding and harmonic feature fusion. *Electronics*, *14*(8). doi: 10.3390/electronics14081574

Makonin, S. (2017). *RAE: The Rainforest Automation Energy Dataset.* Harvard Dataverse. doi: 10.7910/DVN/ZJW4LC

Makonin, S., Ellert, B., Bajic, I. V., & Popowich, F. (2016). Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014. *Scientific Data*, *3*(160037), 1–12.

Makonin, S., Popowich, F., Bajić, I. V., Gill, B., & Bartram, L. (2016). Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring. *IEEE Transactions on Smart Grid*, *7*(6), 2575-2585. doi: 10.1109/TSG.2015.2494592

Massidda, L., Marrocu, M., & Manca, S. (2020). Non-intrusive load disaggregation via a fully

convolutional neural network: improving the accuracy on unseen household. In *2020 2nd ieee international conference on industrial electronics for sustainable energy systems (ieses)* (Vol. 1, p. 317-322). doi: 10.1109/IESES45645.2020.9210661

Medico, R., Baets, L. D., Gao, J., Giri, S., Kara, E., Dhaene, T., ... Deschrijver, D. (2020, 1). PLAID - A Voltage and Current Measurement Dataset for Plug Load Appliance Identification in Households.
doi: 10.6084/m9.figshare.10084619.v1

Moreno, S., Teran, H., Villarreal, R., Vega-Sampayo, Y., Paez, J., Ochoa, C., ... Montoya, C. (2024). An ensemble method for non-intrusive load monitoring (nilm) applied to deep learning approaches. *Energies*, *17*(18). doi: 10.3390/en17184548

Mulinari, B. M., de Campos, D. P., da Costa, C. H., Ancelmo, H. C., Lazzaretti, A. E., Oroski, E., ... Linhares, R. R. (2019). A new set of steady-state and transient features for power signature analysis based on vi trajectory. In *2019 ieee pes innovative smart grid technologies conference-latin america (isgt latin america)* (pp. 1–6). doi: 10.1109/ISGT-LA.2019.8895360

Murray, D., Stankovic, L., & Stankovic, V. (2017). An electrical load measurements dataset of united kingdom households from a two-year longitudinal study. *Scientific Data*, *4*, 160122. doi: 10.1038/sdata.2016.122

Mylona, D. N., & Bouhouras, A. S. (2025). A digital twin-based framework for load identification using odd harmonic current plots. *Applied Intelligence*, *55*(7), 1–15.

Narges Zaeri Esfahani, H. B. G., Araz Ashouri, & Bahiraei, F. (2024). Energy consumption disaggregation in commercial buildings: a time series decomposition approach. *Science and Technology for the Built Environment*, *30*(6), 660–674. doi: 10.1080/23744731.2024.2304539

Nie, Z., Yang, Y., & Xu, Q. (2022). An ensemble-policy non-intrusive load monitoring technique based entirely on deep feature-guided attention mechanism. *Energy and Buildings*, *273*, 112356. doi: https://doi.org/10.1016/j.enbuild.2022.112356

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Picon, T., Nait Meziane, M., Ravier, P., Lamarque, G., Novello, C., Le Bunetel, J.-C., & Raingeaud, Y. (2016). COOLL: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification. *arXiv preprint arXiv:1611.05803 [cs.OH]*.

Rafiq, H., Manandhar, P., Rodriguez-Ubinas, E., Ahmed Qureshi, O., & Palpanas, T. (2024). A review of current methods and challenges of advanced deep learning-based non-intrusive load monitoring (nilm) in residential context. *Energy and Buildings*, *305*, 113890. doi: https://doi.org/10.1016/j.enbuild.2024.113890

Rafiq, H., Shi, X., Zhang, H., Li, H., & Ochani, M. (2020, 05). A deep recurrent neural network for non-intrusive load monitoring based on multi-feature input space and post-processing. *Energies*, *13*, 2195. doi: 10.3390/en13092195

Reinhardt, A., Burkhardt, D., Zaheer, M., & Steinmetz, R. (2012). Electric appliance classification based on distributed high resolution current sensing. In *37th annual ieee conference on local computer networks - workshops* (p. 999-1005). doi: 10.1109/LCNW.2012.6424093

Roos, J., Lane, I., Botha, E., & Hancke, G. (1994). Using neural networks for non-intrusive monitoring of industrial electrical loads. In *Conference proceedings. 10th anniversary. imtc/94. advanced technologies in i & m. 1994 ieee instrumentation and measurement technolgy conference (cat. no.94ch3424-9)* (p. 1115-1118 vol.3). doi: 10.1109/IMTC.1994.351862

Shang, R., Chen, S., Chen, Z., & Lu, C.-T. (2024). Graphnilm: A graph neural network for energy disaggregation. In D.-N. Yang, X. Xie, V. S. Tseng, J. Pei, J.-W. Huang, & J. C.-W. Lin (Eds.), *Advances in knowledge discovery and data mining* (pp. 431–443). Singapore: Springer Nature Singapore.

Song, J., Wang, H., Du, M., Peng, L., Zhang, S., & Xu, G. (2021, 01). Non-intrusive load identification method based on improved long short term memory network. *Energies*, *14*, 684. doi: 10.3390/en14030684

Srinivasan, D., Ng, W., & Liew, A. (2006). Neural-network-based signature recognition for harmonic source identification. *IEEE Transactions on Power Delivery*, *21*(1), 398-405. doi: 10.1109/TPWRD.2005.852370

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013, 17–19 Jun). On the importance of initialization and momentum in deep learning. In S. Dasgupta & D. McAllester (Eds.), *Proceedings*

of the 30th international conference on machine learning (Vol. 28, pp. 1139–1147). Atlanta, Georgia, USA: PMLR.

Todic, T., Stankovic, V., & Stankovic, L. (2023). An active learning framework for the low-frequency non-intrusive load monitoring problem. *Applied Energy*, *341*, 121078. doi: https://doi.org/10.1016/j.apenergy.2023.121078

Wang, S., Chen, H., Guo, L., & Xu, D. (2021). Non-intrusive load identification based on the improved voltage-current trajectory with discrete color encoding background and deep-forest classifier. *Energy and Buildings*, *244*, 111043. doi: https://doi.org/10.1016/j.enbuild.2021.111043

Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. doi: 10.21105/joss.03021

Wu, Z., Wang, C., Xiong, L., Li, R., Wu, T., & Zhang, H. (2023). A smart socket for real-time nonintrusive load monitoring. *IEEE Transactions on Industrial Electronics*, *70*(10), 10618-10627. doi: 10.1109/TIE.2022.3224164

Xiang, Y., Ding, Y., Luo, Q., Wang, P., Li, Q., Liu, H., ... Cheng, H. (2022). Non-invasive load identification algorithm based on color coding and feature fusion of power and current. *Frontiers in Energy Research*, *10*, 899669.

Yan, Z., Hao, P., Nardello, M., Brunelli, D., & Wen, H. (2025). A generalizable load recognition method in nilm based on transferable random forest. *IEEE Transactions on Instrumentation and Measurement*, *74*, 1-12. doi: 10.1109/TIM.2025.3570355

Yaniv, A., & Beck, Y. (2024). Enhancing nilm classification via robust principal component analysis dimension reduction. *Heliyon*, *10*(9), e30607. doi: https://doi.org/10.1016/j.heliyon.2024.e30607

Zeifman, M., & Roth, K. (2011). Nonintrusive appliance load monitoring: Review and outlook. *IEEE Transactions on Consumer Electronics*, *57*(1), 76-84. doi: 10.1109/TCE.2011.5735484