Smart Metering System: Developing New Designs to Improve Privacy and Functionality

Xiaoyu Zhang

Power System Group

Department of Electronic Engineering

Royal Holloway, University of London

This thesis is submitted for the degree of

Doctor of Philosophy

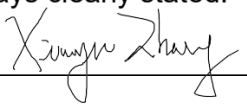
Jun, 2022

Declaration of Authorship

The title page should be followed by a signed declaration that the work presented in the thesis is the candidate's own. Please note that there is no set wording for this but an example is provided below:

Declaration of Authorship

I Xiaoyu Zhang …. (please insert name) hereby declare that this thesis and the work presented in it is entirely my own. Where I have consulted the work of others, this is always clearly stated.

Signed: 

Date: 11st/Oct/2021

Acknowledgements

First and foremost, I am extremely grateful to my supervisor Dr Stefanie Kuenzel for her invaluable supervision, support and tutelage during the course of my PhD degree. She is the kindest, and most patient person I have ever met, it's my great honour to work with her.

Additionally, I would like to express gratitude to my co-supervisors Prof. Chris Watkins and Dr José-Rodrigo Córdoba-Pachón for their treasured support which was influential in shaping my experiment methods in deep learning and data ethics. Chris and Jose are humorous and kind people, and they always inspired me during our talks.

My gratitude extends to The Leverhulme Trust and Royal Holloway, University of London, who jointly funded my PhD project and provided me with the opportunity to undertake my PhD studies, this work would not be possible without assistance from them.

I would like to thank all my colleagues and research teammates in Shilling Building who have offered me generous help and assistants. Particularly, I would like to thank Dr Clive Cheong Took, for his generous advice on machine learning and signal processing.

I would also want to thank my housemates: Mr Jiabao Sun, Mr Xingtai Chen, Miss. Qi An, and Miss. Yunting Qi for a cherished time spent together in the house and the wonderful moment when we were driving a car chasing Aurora in Arctic Circle.

My appreciation also goes out to my parents, my grandmother and friends for their encouragement and support all through my studies from ten thousand kilometres away.

Last but not least, I would like to express my gratefulness for the hardest days I was working during the COVID-19 pandemic, this experience helped me become a more brave, confident, and tenacious person, and enables me better to prepare for the next stage of life.

Abstract

This PhD project aims to develop a novel smart metering system that plays a dual role: Fulfil basic functions (metering, billing, management of demand for energy in grids) and protect households from privacy intrusions whilst enabling them a degree of freedom. The first two chapters of the thesis will introduce the research background and a detailed literature review on state-of-the-art works for protecting smart meter data. Chapter 3 discusses theory foundations for smart meter data analytics, including machine learning, deep learning, and information theory foundations. The rest of the thesis is split into two parts, 'Privacy' and 'Functionality', respectively. In the 'Privacy' part, the overall smart metering system, as well as privacy configurations, are presented. A threat/adversary model is developed at first. Then a multi-channel smart metering system is designed to reduce the privacy risks of the adversary. Each channel of the system is responsible for one functionality by transmitting different granular smart meter data. In addition, the privacy boundary of the smart meter data in the proposed system is also discovered by introducing a data mining algorithm. By employing the algorithm, a three-level privacy boundary is concluded. Furthermore, a differentially private federated learning-based value-added service platform is designed to provide flexible privacy guarantees to consumers and balance the trade-off between privacy loss and service accuracy. In the 'Functionality' part, three feeder-level functionalities: load forecasting, solar energy separation, and energy disaggregation are evaluated. These functionalities will increase the predictability, visibility, and controllability of the distributed network without utilizing household smart meter data. Finally, the thesis will conclude and summarize the overall system and highlight the contributions and novelties of this project.

Contents

Word count: 85,230

List of Figures

List of Tables

Nomenclature

AMI	Advanced Metering Infrastructure
AACE	Absolute Average Coverage Error
AC	Air Conditioner
Adam	Adaptive Moment Estimation
AI	Artificial Intelligence
BLSTM	Bidirectional Long Short-Term Memory
BTM	Behind-the-Meter
CC	Communication Cost
CNN	Convolutional Neural Network
CPC	Computation Cost
CR	Communication Round
DCC	Data and Communications Companies
DL	Deep Learning
DHI	Diffuse Horizontal Irradiance
DNI	Direct Normal Irradiance
DNN	Deep Neural Network
DP	Differential Privacy
DPFL	Differentially Private Federated Learning
DR	Demand Response
DSM	Demand-Side Management
DWT	Discrete Wavelet Transform
EV	Electric Vehicle

ES	Energy Supplier
FL	Federated Learning
GAN	Generative Adversarial Network
GB	Gradient Boosting Machine
GBQR	Gradient Boosted Quantile Regression
GDP	Global Differential Privacy
GDPR	General Data Protection Regulation
GRU	Gated Recurrent Unit
GHI	Global Horizontal Irradiance
GM	Gaussian Mechanism
GSP	Grid Supply Point
HAN	Home Area Network
HE	Homomorphic Encryption
HVAC	Heating, Ventilation and Air conditioning
KNN	K-Nearest Neighbours
LDP	Local Differential Privacy
LSTM	Long Short-Term Memory
MI	Mutual Information
MLP	Multi-Layer Perceptron
MO	Microwave Oven
MPC	Multi Party Computation
MQRNN	Multi-Quantile Recurrent Neural Network
MQ-GRU	Multi-Quantile Gated Recurrent Unit
MQ-LSTM	Multi-Quantile Long Short-Term Memory
MQ-CNN	Multi-Quantile Convolutional Neural Network
NCEI	National Centres for Environmental Information
NILM	Nonintrusive Load Monitoring
NO	Network Operator
Non-TCL	Non-Thermostatically Controlled Load

nMAE	Normalized Mean Absolute Error
Ofgem	The Office of Gas and Electricity Markets
OL	Other Loads
PbD	Privacy by Design
PCA	Principal Component Analysis
PCC	Point of Common Coupling
PIs	Prediction Intervals
PICP	Prediction Interval Coverage Probability
PINC	Prediction Interval Nominal Confidence
PPDL	Privacy-Preserving Deep Learning
PV	Photovoltaic
Q-GBRT	Quantile Gradient Boosting Regression Tree
Q-LGB	Quantile LightGBM
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SCADA	Supervisory Control and Data Acquisition
SGD	Stochastic Gradient Descent
STLF	Short-Term Load Forecasting
TCL	Thermostatically Controlled Load
TOU	Time-of-Use
TTP	Trusted Third Parties
TP	Third Party
TPS	Third Party Service
WAN	Wide Area Network
WS	Winkler Score

Part I Background

Chapter 1 Introduction

1.1 Introduction

This chapter starts by introducing the research background and motivation of the thesis. Then research questions and objectives are determined. The rest of this chapter illustrates the methodology and contributions, and the publications list is presented at the end of this chapter.

1.2 Research Background

The smart grid is a worldwide modernization of electrical power systems in the 21st century. Two-way communication networks enable smart grids to collect real-time data from both the electricity supply (i.e., power stations) and demand (i.e., households) sides and further boost the power system's reliability, availability, and efficiency.

As an essential enabler and prerequisite of the smart grid, smart meters are being installed country- and worldwide at single houses to collect real-time data on energy consumption. As shown in Figure 1-1, a steady increase in the shipment volume of the smart meters is estimated from 2018 to 2024. The overall shipment of the smart meters is expected to surpass 200 million in 2024, which will increase by 35% since 2018 [1]. In North American countries such as the United States and Canada, the smart meter market is well-developed, and around 30-40% of consumers have already installed the smart meters. In Europe, driven by large roll-out plans, the smart meter penetration rate is also growing and approaching maturity. As for the Asia-Pacific

region, the emerging markets led by China and India make Asia-Pacific the largest region in global shipment volume. In 2018, the number occupied 60% of the overall volume.

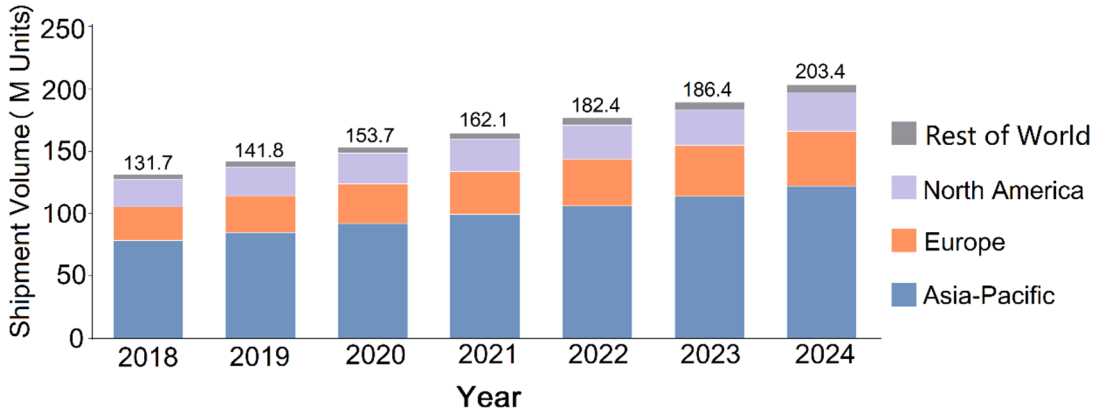


Figure 1-1. Global smart meter shipment volume by region (Adopted from [1]).

However, with the large-scale roll-out plans worldwide, worries about privacy intrusions caused by smart meters are rising as well. Researchers point out that private household information can be revealed by smart meters [2-4]. Through continuously monitoring the real-time smart meter data, the adversaries could have an inside view of household activities and behaviours (e.g., how many residents live in the house, when people leave home, what the residents are doing at particular durations, such as sleeping, bathing, watching TV, washing clothes, etc.). Although data collection may be justified on ethical grounds of utilitarianism (i.e., ensuring the greater, collective good of energy efficiencies in smart grids), the intrusion into privacy could also have negative ethical and social consequences, including the conditional shaping of freedom and behaviour of individuals and households [5, 6].

At a legal level, the General Data Protection Regulation (GDPR) has been in force since 25th May 2018 [7]. Covering all European countries, the purpose of GDPR is to protect all EU citizens from privacy and data violation, providing more power to individuals to control their personal information. With these operational and legal operational possibilities, it is also important to consider ‘soft’ ethical strategies that use them to contribute to protecting household privacy, potentially enabling

households to be more in control of their digital data [8]. One such strategy is considering different stakeholders involved or affected by digital data gathering [9].

Given the scale of smart meter roll-out processes in countries and worldwide, the above risks and operational strategies could be dismissed or subordinated to utilitarian market logic, with the responsibility for their implementation and subsequent privacy protection of consumers (i.e., households) delegated to third parties, many of whom might not have privacy protection as a priority in their agendas. Moreover, and as stated before, there is a lack of clarity about such responsibilities. Furthermore, whilst smart grids could be conceived as necessary technologies to regulate the conduct of individuals in the societies [10], what could be more concerning is that privacy intrusion could also generate negative social consequences [11]. Consumers can be left powerless or socially isolated to devise strategies to counteract intrusion into their privacy, becoming mere means rather than ends [5].

1.3 Potential beneficiaries of the research

The potential beneficiaries of this research involve energy consumers, energy utilities/ third-party service providers, industry/smart meter manufacturers, and regulators/policy makers.

1) **Energy consumer:** The energy consumer is the biggest beneficiary of the proposed privacy-preserving smart metering system. Firstly, referring to the guidance included in this research, energy consumers can be better aware of the potential privacy risks the smart meter brings; secondly, the proposed user-centric smart metering system can provide the consumers with enough personal autonomy and freedom to control their energy data.
2) **Energy utilities/ third-party service providers**: Energy utilities include Energy Supplier (ES) and Distribution Network Operator (DNO). ES is ultimately responsible for rolling out the smart meters across the U.K. at their own expense. Whilst the existing technical solutions require extra energy storage or a high computational server, the low-cost technical solution proposed in this research

helps ES save the privacy budget and actively engages the ES to participate in the smart meter data protection plan. DNO is the operator and management of the distribution network; the energy data from the domestic smart meters help the DNO increase the visibility of the Low-Voltage (LV) distribution network with the high penetration of distributed renewable energy generations. However, the access to smart meter data from the DNO is strictly limited by the data access framework published by the Department for Business, Energy & Industrial Strategy (BEIS) in the U.K. [12], which is the main barrier to the development of grid management and operation applications at the distribution level. The DNO can benefit from this research by increasing the distribution network's visibility, predictability, and controllability with the aggregated smart meter data, while the individual smart meter is kept confidential. Third-party service providers represent the commercial companies that would like to access the consumers' energy data to provide commercial services, e.g., energy data analytics and load forecasting. Typically the value-added services require consumers to submit their energy data to a server, which conflicts with BEIS's data access framework [12]. The third-party service providers can benefit from this research by designing an edge-cloud computing service platform to implement cloud analytics without collecting individuals' data.

3) **Industry/smart meter manufacturers:** The industry/smart meter manufacturers can develop new designs of the smart meter device to better fit the computation, communication, and storage capacity suggested in this thesis.
4) **Regulators/policy makers**: Regulators/policy makers may use this research to design data regulations and laws aligned with the smart grid need.

1.4 Problem Statement

Can we develop a smart metering system design that fulfils reasonable and ethical user and system functionality whilst protecting user privacy and ensuring consent?

1) How does smart meter data reveal consumers' private information, and to whom?
2) Who are the adversaries/attackers, and with what purpose? Moreover, how the proposed system reduces privacy threats?
3) How can the proposed smart metering improvements influence standards, governments and companies for the roll-out, management and use of smart meters?
4) How are the critical functions realized in the proposed smart metering system?

1.5 Objectives

Based on the knowledge gaps discussed in Subsection 1.3.1 and the problems stated above, the objectives of the thesis are listed as follows:

1) Understand the available smart meter technologies, how they work, and the different configurations, options and limitations for setting them up and managing them by both companies and households.
2) Develop the adversary/attacker model, which tries to infer personal information from the smart meter/ advanced metering infrastructure.
3) Develop a privacy-preserving smart metering system with various configuration/design options, which play a dual role: Fulfil basic functions (metering, billing, management of demand for energy in grids) and protect households from privacy intrusions whilst enabling them a degree of freedom.
4) Investigate architectural system options for smart meter data analysis and mining, satisfying load forecasting, balancing consumer behaviour and privacy visibility and allowing for home solar generation.
5) Investigate the influence of the proposed scheme on the industrial specifications.

1.6 Overview of Methodology

The methodology overview is summarized in the block diagram shown in Figure 1-2. This thesis starts with defining the privacy threat/adversary model by identifying their purpose, motivation, and the route to obtain sensitive information. Then based on the defined adversary model, GDPR, and compulsory functionalities required by the stakeholders, a privacy-functionality trade-off strategy is developed. Following the strategy, a multi-channel smart metering system is proposed; the system contains three communication channels: a high-frequency aggregation channel, a low-frequency Time-of-Use (TOU) billing channel, and a third-party value-added services channel. Then the proposed system's privacy boundary (aggregation size and interval resolution) is detected by employing a data mining algorithm used by the adversary.

Then the architectural system options for smart meter data analysis and mining are investigated thoroughly. In the turn of the energy suppliers, the consumer's energy consumption and bill data will be stored locally and only shares with the energy supplier at the end of the reporting period. As the honest-but-curious adversary, the third-party service provider is strictly limited to accessing personal information by introducing a federated learning cloud platform; only the model parameters rather than sensitive information are shared with the third parties. As for DNO, the readings from the neighbouring smart meters are aggregated by the physical/informatic aggregator to remove the individual identity before transmitting it with the DNO. Then the DNO utilizes the aggregated data to increase the predictability (load forecasting), the visibility (distribution-level energy disaggregation), and reduce the uncertainty (renewable energy detection at grid supply point) of the distribution network.

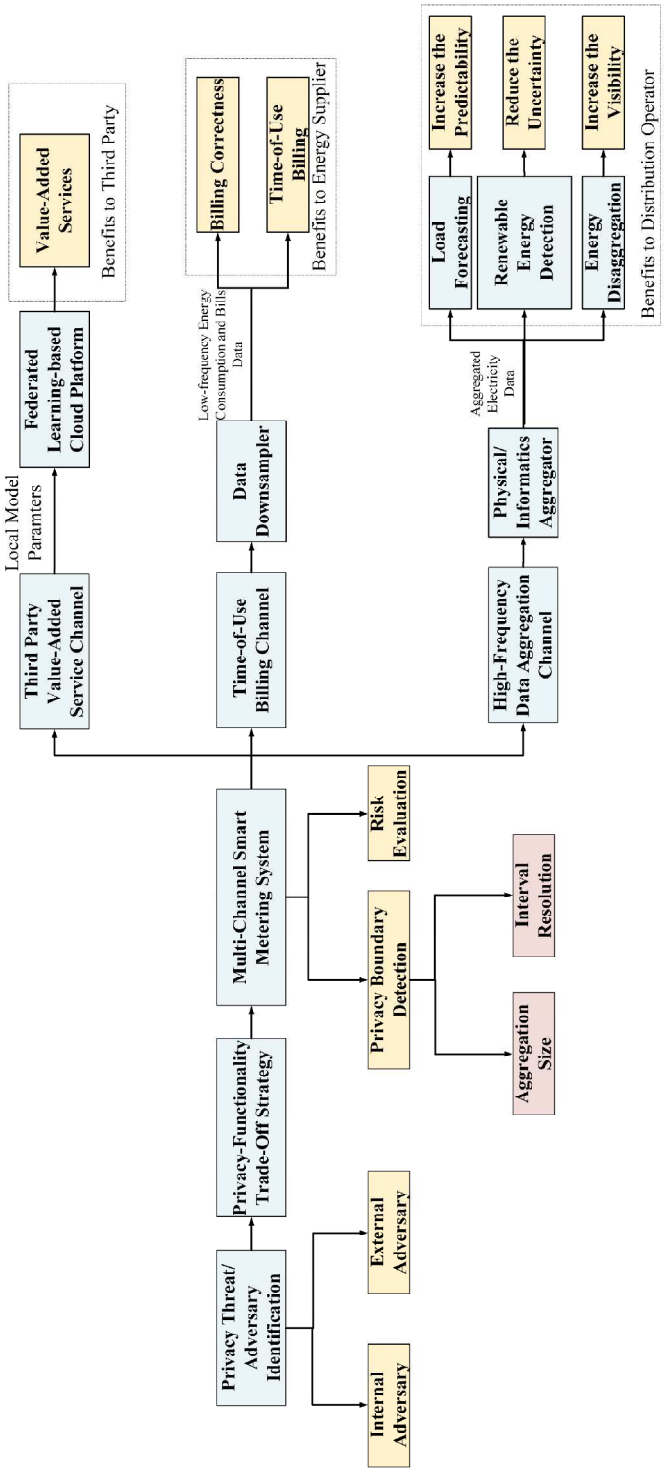


Figure 1-2. Block diagram of the methodology of the thesis.

1.7 Thesis Structure

The rest of the thesis is divided into four parts: Part I Background, Part II Privacy, Part III Functionality, and Part IV Conclusion, respectively.

- **Part I – *Background*** includes Chapter 1, 2 and 3, which introduces the research background of the PhD thesis.
  - **Chapter 1 – *Introduction*** introduces the research background, motivation, research questions, objective, and methodology of the thesis.
  - **Chapter 2 – *Literature Review*** presents a comprehensive literature review is presented. The review includes the existing smart meter technologies, the privacy and function configurations of the current smart metering system, and state-of-the-art research to protect smart meter data. Furthermore, relevant data regulation policies and data ethics knowledge is reviewed.
  - **Chapter 3 – *Smart Meter Data Analytics Methodology*** introduces the theoretical foundations for smart meter data analytics, including machine learning, deep learning, and information theory foundations.
- **Part II – *Privacy*** contains Chapter 4 and 5, which illustrate the privacy-preserving smart metering system configurations.
  - **Chapter 4 – *A Privacy-Preserving Multi-Channel Smart Metering System*** proposes a privacy-preserving smart metering system that combines existing data aggregation and data down-sampling mechanisms. Moreover, the privacy boundary of the smart meter data is detected via an artificial intelligence adversary.
  - **Chapter 5 – *Differentially Private Federated Learning-based Value-Added Service Platform*** develops a value-added service platform based on differentially private federated learning to better balance the services' quality and ensure users' privacy.
- **Part III – *Functionality*** demonstrates the essential functionalities for grid operation and management for LV distribution network purposes that can be realized with the aggregated smart meter data without privacy concerns.

— **Chapter 6 – *Day-Ahead Distribution-Level Spectral Load Forecasting with Aggregated Smart Meter Data*** proposes a hybrid component decomposition and deep neural network day-ahead load forecasting model to fully use both the time domain and frequency domain features of the load demand.

— **Chapter 7 – *A Feeder-Level Solar Energy Decoupling Scheme with Aggregated Smart Meter Data*** introduces an online solar energy decoupling scheme to separate the solar energy generated by the roof-top PV systems from the netload measured at the feeder's head.

— **Chapter 8 – *Multi-Quantile Recurrent Neural Network for Distribution-Level Probabilistic Energy Disaggregation with Aggregated Smart Meter Data*** presents a feeder level probabilistic energy disaggregation model based on a multi-quantile recurrent neural network. The model's target is to disaggregate the demand load into Thermostatically Controlled Loads (TCLs), Non-Thermostatically Controlled Loads (non-TCLs), and non-controllable loads.

- **Part IV – *Conclusion*** contains Chapter 9; this part summarizes the key results and concludes the research. Future research opportunities and potential technology development directions are also discussed.

1.8 List of Publications

The following overview lists the published/submitted journal/conference articles during the postgraduate study:

1.8.1 Journal paper

- **Published journal papers:**

[1] **Zhang, X.Y.**, Kuenzel, S., & Watkins, C. (2022). Multi-Quantile Recurrent Neural Network for Feeder-Level Probabilistic Energy Disaggregation Considering Roof-Top Solar Energy. ***Engineering Applications of Artificial Intelligence***. https://doi.org/ 10.1016/j.engappai.2022.104707

[2] **Zhang, X.Y.**, Kuenzel, S., Colombo, N., & Watkins, C. (2022, Early Access). A Hybrid Short-Term Load Forecasting Method Based on Empirical Wavelet Transform and Bidirectional Long Short-Term Memory Neural Networks. ***Journal of Modern Power Systems and Clean Energy***.

[3] **Zhang, X.Y.**, Watkins, C. , Yin L. and Kuenzel, S. (2022, Accepted with Minor Correction), A Data-Driven Online Solar Energy Disaggregation System from the Grid Supply Point. ***Complex & Intelligent System***.

[4] **Zhang, X.Y.**, Watkins, C., Cheong Took, C., & Kuenzel, S. (2021). Privacy Boundary Determination of Smart Meter Data Using an Artificial Intelligence Adversary. ***International Transactions on Electrical Energy Systems***. https://doi.org/10.1002/2050-7038.13020

[5] **Zhang, X.Y.**, Kuenzel, S., Córdoba-Pachón, J-R., & Watkins, C. (2020). Privacy-Functionality Trade-off: A Privacy-Preserving Multi-Channel Smart Metering System. ***Energies***, 13(12), 1-30. [3221]. https://doi.org/10.3390/en13123221

- **Submitted journal papers:**

[6] **Zhang, X.Y.**, Córdoba-Pachón, J.R., Watkins, C, Kuenzel, S. (Second Round Revision) Differentially Private Federated Learning for Privacy-Preserving Value-Added Services in Advanced Metering Infrastructure. Submitted to ***IEEE Transactions on Computational Social Systems*** in December 2021.

[7] Gao, H., Kuenzel, S., **Zhang, X.Y.** (First Round Revision) A Hybrid ConvLSTM -based Anomaly Detection Method for Combating Energy Theft. Submitted to ***IEEE Transactions on Instrumentation and Measurement*** in April 2022.

[8] Guo, P., Yuan, Z., **Zhang, X.Y.**, Liu, G., Zhao, Y., Kuenzel, S. (Under Review) Key Techniques of Low-Voltage DC Building Distribution and Utilization System and Implementation. Submitted to ***IEEE Transactions on Power Delivery*** in May 2022.

1.8.2 Conference papers

[9] **Zhang, X.Y.**, Kuenzel, S., & Watkins, C. (2020, September). Feeder-Level Deep Learning-based Photovoltaic Penetration Estimation Scheme. In ***2020 12th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*** (pp. 1-5). IEEE. (Best Paper Award First-Grade Prize of the Conference)

[10] **Zhang, X.Y.**, & Kuenzel, S. (2020, October). Differential Privacy for Deep Learning-based Online Energy Disaggregation System. In ***2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*** (pp. 904-908). IEEE. (Participated as the session chair of "Sensors, advanced metering, data acquisition" regular session)

1.8.3 E-Handbook

[11] Gao, H., Kuenzel, S., & **Zhang, X.Y.** (2022). Deep learning for countering energy theft. ***Computer Weekly***. https://www.computerweekly.com/ehandbook/Royal-Holloway-Deep-learning-for-countering-energy-theft

Chapter 2 Literature Review

This chapter presents a wide-ranging literature review of related works in the existing smart metering system, state-of-the-art privacy-preserving technologies, and corresponding applications of the smart meter data. The chapter begins by describing the smart grid technique and advanced metering infrastructure. Then the privacy intrusion issues regarding the smart meter data are investigated. Advanced privacy-preserving techniques that have been applied to protect consumers' privacy are then introduced. Next, the data regulation/laws and data ethics which provide a guideline for smart metering system designing, are presented. Finally, advanced applications of smart meter data and the privacy issues related to these applications are discussed.

2.1 Smart Grid and Advanced Metering Infrastructure

2.1.1 Smart grid

Smart grids are physical networks that use technologies and equipment to interconnect different components through two-way networks that could achieve real-time optimizations to deliver electricity more reliably and efficiently. Smart grids contain not only electricity interfaces but also communication interfaces. Other stakeholders (utility companies) or domains (electricity markets) can be included for analysis and management. Future smart grids can enable better operation and control, better network planning and maintenance, Advanced Smart Metering Infrastructure (AMI), and overall energy efficiency for countries [13].

2.1.2 Smart metering equipment technical specifications

The smart meter is the most important and fundamental device in AMI; there are two types of smart meters in Great Britain: the first-generation smart meter, known as

Smart Meter Equipment Technical Specifications (SMETS) 1, and SMETS 2, which were rolled out since 2018. By the end of 2020, 23.6 million smart meters will have been installed in the UK, while 15.7 million are SMETS1 and 6.7 million are SMETS2 [14]. SMETS 2 has a more advanced communication network which enables the consumers to switch energy suppliers without making the smart meter become a 'dumb' meter; a 'dumb' meter represents the new energy supplier that cannot operate it, and the meter operates like a traditional electricity meter. The sampling frequency of SMETS 2 must be transmitted across the Home Area Network (HAN) at a frequency better than 10s, and it is supposed to achieve a frequency better than 5s in the future [15]. SMETS 2 include monitoring power outages, connecting/ disconnecting the electricity supply, and providing TOU tariffs [15]. In this subsection, the advantages of the smart meter are highlighted by comparing it with traditional electricity meters, and then the physical components of the new generation SMETS 2 are introduced.

2.1.2.1 Comparison among electromechanical electricity meters, automatic meter reading devices, and smart meter

In the past decades, the consumers have witnessed the evolution of electricity meters from electromechanical electricity meters before the 1970s to Automatic Meter Reading (AMR) devices between 1970 and 2000, then to smart meters nowadays. A comparison of these three generations' electricity meters is made in Table 2-1. The electromechanical electricity meter is the major electricity meter during the last century and can only measure the active energy consumption in kWh. The components of this purely mechanical drive device are the driving system, moving system, braking system, and registering system [16]. The principle of the electromechanical electricity meter is simple, as an aluminium disc rotates at speed proportional to the power disc speed, the active energy consumption is computed by counting the revolutions of the aluminium disc. Although the structure of the electromechanical electricity meter is simple and the cost of the device is cheap, there are several main drawbacks of this conventional meter: Firstly, the metal components inside the meter are affected by the

environment and temperature variations easily, susceptible to errors occur as a result. Secondly, the aluminium disc may rotate fast or slow during nonlinear loads such as energy storage systems [17]. Thirdly, this kind of meter requires manually readings, which increases the cost. Moreover, electromechanical electricity meters cannot detect energy theft in time, resulting in billions of pounds lost yearly [18].

With the development of electronic techniques such as Microprocessor Units (MPUs) and fast Analog-to-Digital Converters (ADCs) in the 1990s, electronic components replace most mechanical parts of the electricity meter. AMR utilizes digital technology to collect the power consumption data and transmit the data to the utility for billing, analysing, or troubleshooting purposes. Normally, an AMR device contains a power supply, microcontroller, Real-Time Clock (RTC), Liquid Crystal Display (LCD) display, and communication ports [19]. AMR can provide near real-time reading with high accuracy compared to the electromechanical electricity meter, and AMR is little affected by the environment [20]. Another significant advantage of the AMR is that no staff from the energy supplier is required to record the energy consumption on-site, which saves the expense. However, AMR only enables one-way communication, which comes from the meter to the utility, while the utility cannot send information to the end-users.

Since the beginning of 21 century, a new generation of the smart meter has been rolled out in North America and Europe; smart meter is an electronic device which is more advanced than AMR. The smart meter can collect more electricity parameters, including phase voltages, phase currents, frequency, power factor, active power, reactive power, apparent power, and power quality measurements. Moreover, a smart meter enables two-way communication between consumers and energy suppliers; the communication can be wireless or wired (such as Powerline Communication (PLC)). Furthermore, the smart meter aims to play the role of a communication hub to provide consumer value-added services such as outage management and customer load management.

Table 2-1. Comparison between the smart meter and traditional electricity meters.

Feature	Electromechanical Meter	Automatic Meter Reading (AMR)	Smart Meter
Communication	No communication	Real-time one-way communication	Real-time two-way communication
Accuracy and reliability	Low	High	Very high
Energy theft detection	Low	At node level	At network level
Time-of-use tariff	Unavailable	Unavailable	Available
Additional device	No	No	In-Home Display (IHD)
Consumer participation	No	Low	High
Business opportunity	Monthly billing	Monthly billing	1. TOU billing 2. Consumer payment option 3. Utility operation 4. Demand response 5. Outage management 6. Information display

2.1.2.2 Physical components

The components of a SMETS 2 smart meter include an electricity meter, a power supply unit, a micro-controller, an RTC, an In-Home Display (IHD), an LCD, a data store module, a load switch, a HAN interface, and a communication hub [20] (see Figure 2-1). The detailed function of each component is introduced as follows [20]:

- **A power supply unit**: the power supply unit supplies power to other hardware components such as the micro-controller and the communication unit in the smart meter, this unit normally contains step-down transformers, rectifiers, AC-DC converters, DC-DC converters and regulators [20].
- **An electricity meter**: The electricity meter contains voltage, current sensors, and an energy measurement unit. The voltage and current sensors collect the input signals, while the Energy Measurement Unit (EMU) is responsible for signal conditioning, ADC, and computation. The EMU outputs active, reactive, and apparent energy consumption.

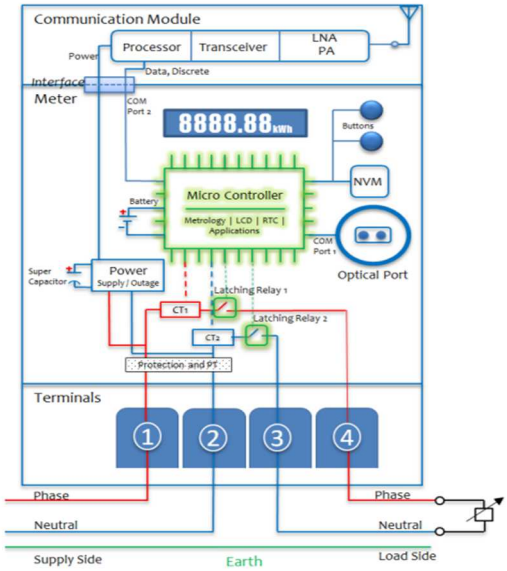


Figure 2-1. Smart meter physical components (Adopted from [20]).

- **An RTC**: RTC is an inbuilt block to ensure the smart meter keeps track of the real-time and avoids time drifts; RTC also provides essential tariff information by dividing the timesteps into tariff slots (time of the day). Referring to Smart Metering Equipment Technical Specifications, the error should be within 10 seconds of the UTC date and time [15].
- **A micro-controller**: the micro-controller is the core component of the smart meter; the micro-controller performs almost all functions. High measuring accuracy and energy efficiency, high degree of parallelism are the basic requirements for the micro-controller, as the micro-controller needs to handle multiple tasks in parallel, including calculation with the data collected, post-processing, data formatting, communication with other communication devices, displaying electrical parameters, tariff, and bills, etc. [17].
- **An IHD**: IHD is a small digital screen which connects the smart meter via HAN. IHD shows energy usage in kilowatt-hours (kWh) and bills in pounds and pence.
- **An LCD**: The LCDs the computed energy consumption for billing purposes.
- **A data store module**: The module contains an inbuilt flash memory card to record the power consumption details with timestamps. A smart meter can typically store around 13 months of half-hourly usage data.

- **A HAN interface**: The HAN can establish a ZigBee SEP v1.2 Smart Metering HAN to support the routing of commands, responses, and alerts to and from devices and support the cryptographic suite [21]. ZigBee is the communication protocol based on IEEE 802.15.4 MAC [22], widely applied in smart meters.
- **A load switch**: an Auxiliary Load Control Switch (ALCS) in the smart meter can switch a second electrical circuit off and on; the switch pattern can be activated by either setting a calendar in the meter, giving the schedule to the consumer or receiving the command from the energy suppliers [23].
- **A communication hub**: The communication hub is the central communication component of the smart meter [24]. It has two functions: Firstly, it enables the smart meter and IHD device to communicate with each other via HAN; secondly, the communication hub plays the role of a gateway to link the HAN with WAN to allow the collected data to be transmitted to the energy supplier, the network operator, and the third-party service providers.

2.1.2.3 Data recording

Data recording is an important function of the smart meter for both utility and consumers. Utilities monitor the status of energy networks and are useful for consumers to know the details about their energy use. Under the SMETS 2 standards regulated by the Department for Business, Energy & Industrial Strategy (BEIS) in the UK [12], although the frequency of data transition is high (greater than 10s), only low-frequency data is recorded in the storage unit on a long-term basis [25]. The smart meter stores four categories: half-hourly data, daily totals consumption data, historical TOU tariff, and other totals consumption and cost data (see Table 2-2). Frist of all, all data is recorded with the timestamp in UTC date and time format. Half-hourly energy consumption data is the highest resolution data stored in the smart meter; 13 months of historical half-hourly energy consumption in kWh is stored, and three months of data of cumulative active energy imported in the active import register, three months of data of cumulative reactive energy imported/ exported in the active import/export register. The daily energy consumption data is the second-highest resolution; two

years (731 days) of data is stored. The cost data in £ in pairs with the energy consumption is also available in the storage module, as shown in Table 2-2.

Table 2-2. Smart meter data storage requirements under the SMETS 2 standards [15, 25].

Parameter	Duration
Half-hourly data	
• Active energy consumption	13 months
• Active energy exported	Three months
• Reactive energy imported	Three months
• Reactive energy exported	Three months
Daily totals consumption	Two years
Other totals consumption (kWh) and cost (£)	
• Daily	Current day plus prior eight days
• Weekly	Current week plus prior five weeks
• Monthly	Current month plus prior 13 months
Time-of-use tariff (£/kWh)	
• Tariff TOU Register Matrix	A 1 x 48 matrix

2.1.3 Advanced smart metering infrastructures

Within smart grids, AMI systems are integrations of smart meters, communication networks, and data management systems (see Figure 2-2) [26, 27]. With advanced communication techniques, AMI enables real-time bidirectional communication between the suppliers and electricity consumers [28]. Smart meters are the most vital components of AMI. As smart energy sensors are installed in consumers' residences (households), smart meters can gather and transmit data, including power consumption and electricity/gas bills, on a real-time basis.

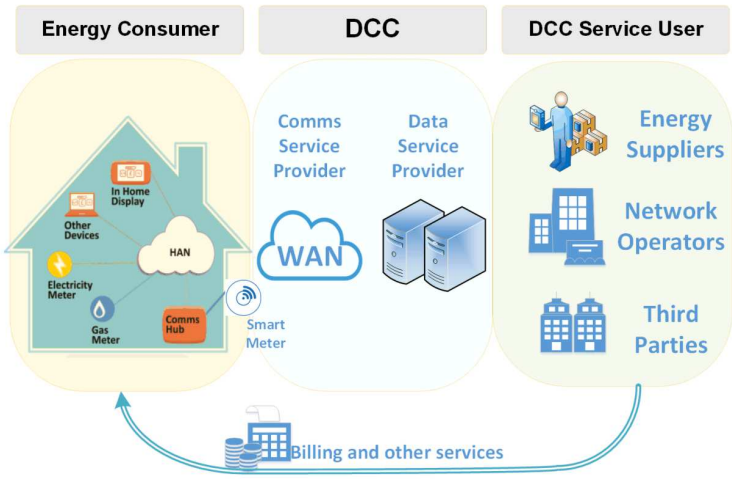


Figure 2-2. The block diagram of the current smart metering system (Adopted from [29]).

Like most Internet of Things (IoT) systems, the smart metering system contains several layers for data acquisition, communication, and computation. Normally, the smart metering system is a four-layered architecture shown in Figure 2-3, and the architecture contains a physical layer, network layer, middleware layer, and application layer.

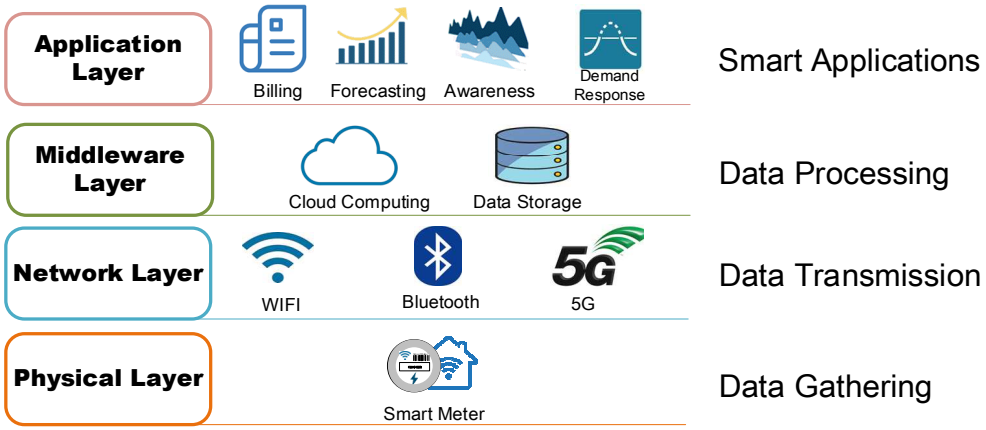


Figure 2-3. The four-layered architecture of the smart metering system (Adopted from [29]).

The first layer is the physical layer, also called the perception layer; the layer's responsibility is to utilize smart sensors (including smart meter and smart plug) to collect data from the consumer's home [30]. The second layer, the network layer, enables the communication between the smart meter and the cloud server and transmits power consumption to energy suppliers. In the third layer, named the middleware layer, the collected data is processed, analysed, and stored here. Finally, the essential applications and services are delivered in the application layer, including load forecasting, energy awareness, time-of-use billing, demand response and so on [31].

2.1.3.1 Communication network

The communication network of AMI is the most critical part of the overall system, enabling two-way real-time communication. With the support of the communication network, the smart meter plays the role of a communication hub to communicate with the energy suppliers, domestic appliances, neighbouring smart meters, and other parties. The hierarchical multi-layer communication network contains three

communication protocols: Wide Area Network, Local Area Network (LAN)/ Neighbourhood Area Network (NAN), and HAN.

2.1.3.1.1 Home area network

HAN is a network inside a customer's home that connects a smart meter to an IHD and other devices (controllable loads, renewable generators, etc.). The communication protocol of the HAN can either be wired (such as PLC) or wireless techniques (including Zig-bee, Z-wave, Wi-Fi, etc.) [20]. Since all domestic appliances are inside the residential buildings, the HAN requires a short converge range (up to 100 kbps) and a low data rate (up to 100 m). the ZigBee at 2.4 GHz is the widest communication protocol employed in HAN since ZigBee is a simple, low-cost, low-power, and secure wireless communication technology [32]. The HAN is connected to the upper communication networks such as LAN or WAN via the smart meter.

2.1.3.1.2 Local area network

LAN/NAN is the middleware network between HAN and WAN; the responsibility of LAN is to transmit data among neighbouring smart meters or transmit data from many smart meters to a data concentrator [33]. Compared to HAN, LAN requires a higher data rate (100 kbps–10 Mbps) and a larger coverage distance (up to 10 km). Long-distance wired/wireless communication techniques such as ZigBee mesh networks, WiFi mesh networks, Worldwide Interoperability for Microwave Access (WiMAX), Cellular, PLC, and Coaxial Cable are employed to implement the LAN/NAN. As an important part of AMI, LAN/NAN ensures the consumption information is transmitted between the smart meters and other stakeholders (energy supplier or third-party service providers); LAN/NAN also support various applications such as remote meter reading, detection of unauthorized usage. LAN/NAN is then connected to WAN through a backhaul network, where the data from many LAN/NAN is concentrated together and transmitted between WAN and LANs.

2.1.3.1.3 Wide area network

The WAN links the smart meters to Data and Communications Company (DCC). WAN covers a large geographic area such as a country or a state; it needs to transmit a large volume of data at a high frequency to enable the real-time stability control of the power system, and the communication technique requires a much higher data rate (10 Mbps–1 Gbps) and long coverage distance (up to 100 km) comparing to HAN and LAN/NAN. The WAN can be developed with either wired technologies (such as broadband power line communication (BPL), fibre optics) or wireless technologies (such as Cellular and WiMAX) [17]; satellite communication is also employed as a backup in some remote areas [33].

2.1.3.2 Data and communications company

As shown in Figure 2-2, DCC is a private central communication sector responsible for secure communication between smart meters and authorised users, access control and scheduled data retrieval [34]; the DCC is licenced by the UK government and under regulation by OFGEM. The DCC collect energy consumers' data through the WAN; the processed data is then sent to energy suppliers and network operators. The DCC is also the security entity responsible for the security during data carriage and translation, ensuring all process compliance with the GDPR directive. Furthermore, the DCC is also responsible for delivering the interoperability, enabling the energy consumers to switch the energy suppliers and change the tariff plans easily with the forthcoming Central Switching Service (CSS) system [35]. It is noticed that not all functions are under the mandate of the DCC; the critical and core services such as billing or outage management are mandated services, while non-mandated services include analysis of measured energy usage data or automatic tariff comparison services, which require the interaction with personal devices such as mobile phones, computers [36].

2.1.3.3 Stakeholders

As illustrated in Figure 2-2, stakeholders of the smart metering system can include the consumers, energy suppliers, network operators and third parties. With smart meters, consumers can obtain near real-time and more accurate power usage data and bills, which helps them manage their energy usage.

2.1.3.3.1 Energy consumer

Energy consumers in the distribution network include residential energy consumers, commercial energy consumers, industrial energy consumers (accounts for 32% of energy use), and transportation (27% of energy use) [37]. The residential energy consumer contains single-family and multi-family houses, which account for 37% of total energy use. Heating, Ventilation and Air Conditioning (HVAC), lighting, and water heating are the electric loads that consume the most in residential energy consumers. The peak load of the residential energy consumers appears in the afternoon and early evening. The commercial consumers include private/commercial companies, government facilities, service-providing facilities and equipment [38]; this sector accounts for 35% of all energy use. The peak demand of the commercial energy sector appears during the operating hours on weekdays, while the energy consumption decreases during evenings and weekends. Industrial energy consumers contain industrial facilities and equipment for manufacturing, mining, agriculture, and construction, this part of energy occupies 27% of the overall energy consumed annually [39]. The machine drive is the load which consumes over half of the energy, and one characteristic of the industrial load is that the load curve does not have significant seasonal trends; the curve will not change throughout the year. This research mainly focuses on the residential energy consumer, and the domestic smart meter, which collects household-level energy consumption; other categories of smart meter and energy consumers are out of the scope of this research.

Referring to the data access and privacy framework published by BEIS in the U.K [12], the energy consumers have considerable flexibility in accessing their power

consumption data through IHD or Consumer Access Devices (CAD) via HAN. In addition, energy suppliers should provide free of charge data for up to 24 months once the energy consumer requests.

2.1.3.3.2 Distribution network operator

DNO is responsible for constructing, maintaining, and operating the distribution network, ensuring the power is delivered to the end-users. In the UK, 14 different DNO regions are managed by six operators: Electricity North West Limited, Scottish and Southern Electricity Networks, Scottish Power SP Energy Networks, UK Power Networks, and Western Power Distribution [40]. DNO also benefits from the smart meter data with different granularities and aggregation levels. Application such as state estimation, Volt and Var Control (VVC) requires access to highly aggregated smart meter data. While applications such as electricity theft detection, fault location, isolation, and service restoration (FLISR), demand-side management also requires household-level smart meter data.

Although the smart meter data help the DNO improve the distribution system's feasibility, cost-effectiveness and efficiency, DNO can also potentially obtain individual load profiles. Both OFGEM and DBEIS published strict data access framework to regulate the data collected by the DNO to ensure all data access actions are subject to compliance with data protection legislation. Moreover, OFGEM, which regulates the monopoly companies which run the gas and electricity networks in the UK, also published an open letter on DNOs' privacy plans for access to smart meter data in 2016 [41]. This open letter highlighted the importance of the household-level smart meter data for better management and efficiently reinforcing the energy networks. OFGEM requires DNOs to provide sufficient information about their privacy plan to decide whether to approve the assessment. The information includes the variable, the format, the purpose (without any commercial use), the period, and the target consumers of the smart meter data to be collected. Moreover, the quantification of the benefits the collected data brings to the power network, smart grid development, and customers should also be provided to OFGEM. OFGEM and

DBEIS also suggest data aggregation and anonymising techniques to remove the individual features from the collected data.

2.1.3.3.3 Energy supplier

ES are price and contract regulators between the DNO and energy consumers; ES buys electricity from the wholesale market and then sells it to energy consumers. Moreover, ES can provide more flexible and personalized service plans to the energy consumers compared to the utility companies, and ES usually sells the energy at a lower rate. The UK'S typical ES companies include British Gas, EDF Energy, E.ON, RWE Npower, Scottish Power and SSE [42].

Referring to DBEIS's data access and privacy framework [12], ES can either access energy consumption data with no detail more than daily with the consumer's consent or data that is more detailed than daily once the ES both obtains the consumer's consent and provides detailed information about how the data will be used to the consumer. In addition, ES companies are encouraged to develop products and services with collected data to improve the consumers' user experience.

2.1.3.3.4 Third parties

In the field of AMI, the term 'third party' (also named energy service companies or value-added service companies in some research) generally refers to no licensed parties, including energy services companies and switching sites which provide value-added services to the consumers. Third parties are not involved in the grid operation and management or supplying electricity directly, but these companies want to provide additional services by accessing the consumers' energy data. Sharing smart meter data with third parties can promote innovation and competition in the energy services market [12].

DBEIS has strict regulations and limitations on third parties accessing the consumers' energy consumption [12]. Third parties are guided to access personal smart meter data via DCC unless the third parties fulfil all privacy safeguards listed by DBEIS: (1)

Before sending the request to DCC, third parties must obtain the consent from the target consumers, and third parties should also provide detailed information about the purpose, interval of the data, duration, etc.

2.2 Privacy Intrusion Issues

Currently, smart metering systems could easily suffer from internal [43] and external attacks [44] and be subject to privacy intrusion [2]. All privacy intrusion issues related to smart meters fall into two categories:

Category (i) Data sensitivity. Personal energy data cannot be measured by a conventional electricity meter. While the traditional electricity meter measures the energy consumption with a low resolution (e.g., one month) and can only provide the energy consumption information in kWh, the smart meter measures the power consumption with a high frequency (ranging from every second to every half hour, and usually every 15 minutes [45]), and more parameters are recorded, such as real-time active/reactive power, voltage, current, TOU tariff, etc. The high granularity data provide adversaries with enough information to intrude on personal information.

Category (ii) Algorithm sensitivity. Advanced algorithms/mechanisms to intrude on privacy-sensitive features that could not be extracted from raw data using traditional data processing mechanisms. With the implementation of smart meters in smart grids to meet the above functions and the increasing development of new services and applications by TP based on big data and artificial intelligence (AI) (e.g., Machine Learning (ML), Deep Neural Network (DNN), cloud computing), more and more sophisticated data could become available [46]. New services to better understand and monitor household behaviour include NILM [47], short-term load forecasting, distributed data mining, and others [13, 46]. These advanced techniques are a double-edged sword for consumers. The benefits espoused to consumers described above (e.g., managing their energy consumption) and adopting a utilitarian ethic (i.e., ensuring the greater, collective good of energy efficiencies in smart grids) must be weighed against potential privacy intrusion risks. Privacy intrusion would mean that

individual and collective freedom is compromised, given that household behaviour would be shaped and constrained by the perceived presence of digital surveillance [8, 9, 11]. Moreover, referring to the US National Institute of Standards and Technology (NIST) guideline NIST IR 7628v2 [48], the above two categories can be divided into four aspects as follows:

2.2.1 Behaviour patterns identification

Behaviour patterns identification belongs to category i; it aims to identify the appliances used. The smart meter and AMI communication network enable the utility and TP to access individual energy data continuously [26]. The high granularity data can reveal information about specific appliances at certain times and locations inside the home. Based on this information, operators can further infer the activities inside the house [48]. Potential usage of the appliance information may include the retailers would adjust the warranty policy or using the information for advertising and marketing purposes.

2.2.2 Real-time surveillance

Real-time surveillance means that by regularly accessing energy data via smart meters, power system operators/TPs can have an overall picture of the activities inside a house and even the entire life cycles of all residents (waking/sleeping pattern, number of residents, when people leave their home). This privacy concern belongs to category ii; the surveillance relies on simple load monitoring with smart meter/ smart sensors or implements advanced techniques such as data mining and machine learning/deep learning algorithms [46, 47]. This information could be abused by hackers and stolen for an illegal purpose [49].

2.2.3 Fraud

Fraud represents the potential risks of modifying personal energy data without authority, either to increase/decrease energy consumption or attribute the energy

consumption to another house [48]. This risk belongs to category ii; the AMI enables more opportunities for adversaries to implement fraud than conventional meters since the vulnerabilities of the real-time communication network could be abused.

2.2.4 Non-grid commercial uses of data

This privacy risk falls into category ii. TP may use the smart meter data to profit from the data; activities include advertising and insurance that are not welcomed by consumers [48]. Companies would sell their products to residents according to the personal preference information revealed by the energy data. Even sensitive information, such as employment information, income, and the number of residents [46, 50-52], can be inferred from energy data with machine learning algorithms. Adversaries can use this information to estimate the income of the target family.

2.3 Related Work for Privacy Intrusion Protection

The state-of-the-art methods dealing with the above smart meter privacy issues can be divided into user demand shaping and data manipulation (Figure 2-4). Both these techniques reduce privacy loss by decreasing the probability of inferring individual appliance signatures from the overall power data [53].

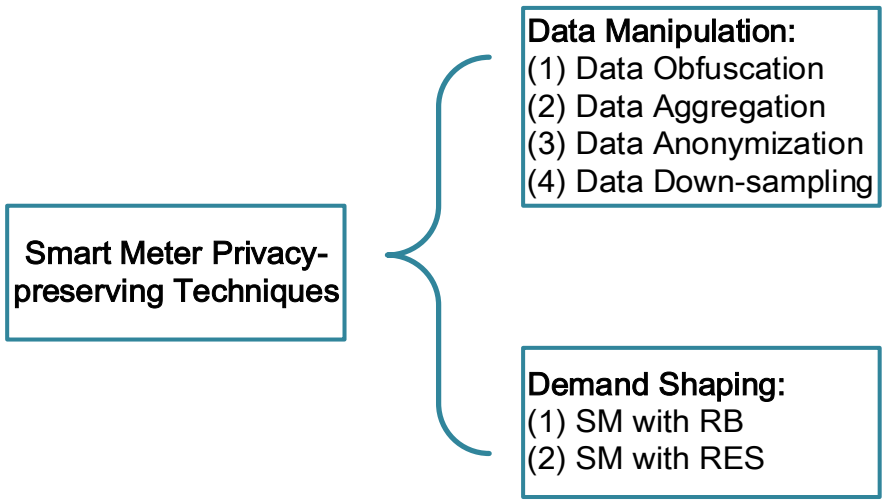


Figure 2-4. Categories of the privacy-preserving techniques.

2.3.1 Demand shaping

User demand shaping uses external energy storage devices (such as a large Rechargeable Battery (RB), Renewable Energy System (RES), or load shifting to distort the actual power consumption curves. The RB and RES method can be treated as a noise-adding approach at the physical layer, as the original power demand is distorted, and the utility cannot infer sensitive information from the smart meter data. An RB system contains a smart meter, a battery, and an Energy Management Unit (EMU). The EMU controls the battery to implement an optimal Energy Management Policy (EMP); with the injection of power from the RB, the mismatching between the power supplied by the grid and consumers' power demand provides a privacy guarantee. The works conclude that the larger the battery capacity, the better the privacy guaranteed. However, the RB is a finite capacity energy storage device with a capacity ranging from 2 kWh to 20 kWh [54]. Therefore, a lower and upper bound exists to limit the mechanism's performance. The optimal EMP, such as the Best-Effort (BE) algorithm [55], water-filling algorithm [56], Q -Learning algorithm [57], and Non-Intrusive Load-Levelling (NILL) algorithm [58], is introduced to optimize the charging/ discharging process. These algorithms control the battery either hide, smooth, or obfuscate the load signature [59]. NILL algorithms are designed to blind the NILM [58]. Instead of only one target load, the NILL has two states, a steady-state and a recovery state; if the battery capacity cannot enable the load to maintain a steady state, the load is switched to the recovery state. A privacy-versus-cost trade-off strategy considering the TOU tariff was proposed by Giaconi et al. in 2017 [60]. Instead of a constant load target, a piecewise load target referring to the current TOU price is generated, the cost of the electricity is minimized, and the consumers can also sell extra energy to the grid to reduce the cost further.

RES utilizes rooftop PV, small wind turbines, and even Electric Vehicle (EVs) [61] to replace conventional batteries. To overcome the difficulty of rolling -out expensive RES and RB facilities, [62] proposed a multiuser shared RSE strategy that enables several users to share one RES and one EMU. The EMU control the RES to allocate the energy from the RES to each user. In this case, the system's target is to minimize

the overall privacy loss of all users rather than an individual user. EV is another scheme to reduce the reliance on the RB [61]. Since the charging period overlaps the peak load, it can mask other appliance signatures. However, the EV can only be used when the consumers are at home, and the consumers are still under real-time surveillance since the adversary would obtain information when the residents leave their homes.

To summarize, in RB/ RES methods, researchers view the identification information of the load curve as the variation of the load measurements of two neighbouring measure points. The ideal situation for the grid curve is a constant value which will not reveal any sensitive features of the demand, and the modified load curve is then compared to the constant value; the more similarities between these two curves, the better privacy can be guaranteed. To quantify the privacy loss, Mean Squared-Error (MSE) [60], Mutual Information (MI), Fisher Information (FI) [55], KL divergence [59], and Empirical MI [63] are adopted in related works. However, user demand shaping also has drawbacks: Firstly, although RB/RES method can provide a certain degree of privacy, the method requires installing extra devices worth thousands of pounds [64]. Who needs to afford this cost is a critical issue and the main barrier to the implementation. An optimal national privacy solution should be a privacy-by-design scheme that does not rely on external devices or facilities. Secondly, the RB harms the environment, which opposes developing countries' carbon peak and carbon neutrality commitments [65]. A more environmental-friendly methods need be proposed.

As the drawbacks of RB/ RES methods are apparent, another demand shaping method named load shifting is proposed to replace the RB/ RES techniques. This method hides sensitive information by shifting the controllable loads [66]. The loads can be divided into uncontrollable loads (e.g., lighting, microvan, kettle) and controllable loads (e.g., HVAC systems, EV, dishwasher, washing machine). Consumers can schedule the operation time and model the controllable loads to prevent occupancy detection. In [67], Combined Heat and Privacy (CHPr) are proposed, and thermal energy storage such as an electric water heater is adopted to mask occupancy. Compared with the RB

approach, CHPr neither requires expensive devices nor increases electricity cost. There are several limitations of the load shifting technique. Firstly, some controllable loads have limited operation modes and cannot be interrupted; secondly, there are restrictions for the thermal loads to store energy.

2.3.2 Data manipulation

Unlike the demand shaping approach, data manipulation aims to modify the smart meter data before sending it to the utility. This category belongs to data aggregation, obfuscation, down-sampling, and anonymization.

Data obfuscation, also called data distortion, tries to add noise to the original smart meter data to cover the actual power consumption [68-70]. Like the demand shaping technique, data obfuscation also reduces privacy loss by distorting the smart meter data on the network layer. Noises such as Gaussian noise [68, 71], Laplace noise [68], and gamma noise [69] are added to the original smart meter data to distort the load curve. These noise-adding mechanisms follow normal distributions with a mean μ equal to zero. Hence the noise would cancel out if enough readings are added up together. P. Barbosa *et al.* [72] conclude that these probability distributions would not influence the relationship between the utility and privacy, so all distributions can achieve similar performance in protecting privacy. Moreover, several methods are proposed to guarantee the billing correctness: [68] proposes a power consumption distribution reconstruction method by adding another Gaussian distribution into the data, but the method does not quantify how much noise should be added to recover the original curve; [72] sends a filtered profile to the utility rather than masked profile, then result shows that the error of the overall power consumption is reduced in this way. However, they also find that the error during different periods (peak period, off-peak period) is significantly different, which provides a new challenge. In summary, although the data distortion scheme effectively reduces privacy loss, several problems should be discussed in future studies: (1) The TOU tariff is unavailable. Although the noise would be zero-mean, the multiplier for TOU pricing is not. Hence the sum of TOU bills would be influenced. (2) The noise-adding process should be an inbuilt

function of the smart meter, which is unrealistic in practice. (2) Although, from the signal processing and information-theoretic viewpoint that a zero-mean noise would not influence the result, it is noticed that the power system is operating on a real-time basis. The power system operator manages the grid with the real-time data sent from the smart meter; even a minor delay or a minor error between the real value and the distorted data could result in serious faults, even the collapse of the whole system. (4) the attacker can remove the noise, which makes the noise-adding method meaningless.

Data aggregation reduces the privacy loss by constructing aggregators to collect the data from a few smart meters together, so the utility cannot detect the electricity events in a single house [71, 73-76]. The data aggregation technique is divided into aggregation methods with Trusted Third Parties (TTP) [71] and aggregation methods without TTP [73, 76]. J.-M Bohli *et al.* [71] propose data aggregation with TTP, the data aggregator operated by the TTP is responsible for gathering the data from neighbouring smart meters and then sending the aggregated data to ES. At the end of every month, the data aggregator also generates the energy consumption of individual consumers for billing purposes. However, there are several concerns about involving TTP. Above all, a TTP has potential motivation to infer personal information, so the TTP itself may bring extra privacy risks to the system. Secondly, with the increasing numbers of smart meters being installed, it is unrealistic for the TTP to build enough data aggregator to satisfy the demand, and the maintenance and development cost would be unaffordable to EP and NO. Thirdly, introducing the data aggregator with TTP decrease the reliability of the system, a single point failure would influence the normal operation of the overall system.

[77-80] introduces data aggregation mechanisms without TTP. Instead, the aggregation process should be combined with other encryption technologies or enforced by law/regulation. Encryption techniques such as Homomorphic Encryption (HE) [77, 79], Multi-Party Computation (MPC) [80], and Block Chain [78] have been employed in the literature. Both HE and MPC encrypt personal smart meter data before sending it to the utility/TP. However, unlike conventional encryption techniques, HE and MPC enable TPs to manipulate the data without knowing the

detail of it. F. Li *et al.* [76] and R. Lu *et al.* [73] independently proposed an aggregation method with HE. By encrypting smart meter data, the data aggregator can implement aggregation without knowing the details. In this way, there are no concerns that the TTP may infer sensitive information without permission. However, the drawbacks of data aggregation technology with HE cannot be ignored. Firstly, after aggregating, the utility cannot obtain the power usage information of an individual consumer. Secondly, HE has very high computation and storage requirements; transmitting energy data with HE would cause a high computational overhead. The computation time to process the cyphertext is about a million times slower than plaintext operations on average [81]. Moreover, considering memory usage, 1 Mb of data results in more than 10 Gb of encrypted data [82]. Considering a national AMI that links millions of smart meters and requires near real-time communication, HE is currently an unacceptable trade-off for utility companies. MPC requires low computing ability but involves several servers to deal with the data [83]. In MPC, each server holds a part of the input data and cannot infer the whole information. MPC has been successfully adopted in smart metering services such as TOU billing. However, complex value-added services, such as load forecasting and online energy disaggregation, require an advanced cloud services to implement these algorithms. So, the availability of MPC in these services should be discussed. The privacy boundary of aggregation size is also investigated in T.N. Buescher *et al.*'s work [74]. They investigated the aggregation size referring to a privacy metric named 'privacy game'. Referring to the data-driven evaluation, a conclusion is made that even a data aggregator with over 100 houses can still reveal private information. But the privacy measure they adopt is abstract and just simply measures the difference between the individual load curve and the aggregated curve, a more detailed privacy measure should be proposed to reflect whether advanced algorithms (such as NILM) can infer personal information from the aggregated data. Another possible data aggregation scheme without TTP is relying on the regulation/law enforced by the government, the measurements from neighbouring smart meters gather at the data aggregator, which is regulated by the government or the DCC, then the DNO or ES accesses the aggregated data via the data aggregator

for controlling and management purpose. To achieve this target, a hierarchical AMI need be developed to include the data aggregators into the smart metering system.

[68-70] combines data aggregation with noise-adding techniques to enable differential privacy of the aggregated data. Differential privacy is employed as a privacy guarantee; differential privacy is through adding noise to a largescale dataset; any two neighbouring datasets (only one data in these two datasets is different) should be indistinguishable [84]. In other aggregation mechanisms, N smart meters are aggregated at first, and then a Distributed Laplacian Perturbation Algorithm (DLPA) is applied to the aggregated data. By adjusting the parameters ε and δ, then (ε, δ)-differential privacy is achieved (ε is the parameter to show the strength of privacy guarantee, and the δ is the failure probability, the closer ε and δ to 0, the better privacy can guarantee). The security and privacy performance are analysed in [68]; two denoising filter attacks, the linear mean filter and the non-local mean filter, are employed to evaluate the original. The results support the point that attackers cannot infer the original load curve from the distorted one.

The Data Anonymization mechanism [85-87] reduces privacy loss by replacing the real smart meter identification with pseudonyms. C. Efthymiou and G. Kalogridis proposed a data anonymization method with a TTP escrow in 2010 [85]. They suggested that two IDs are attached to each smart meter, LFID for sending attributable low frequency and HFID for sending anonymous high-frequency data, while the HFIDs are kept by a TTP, making it unknown to the utility. The low-frequency data are used for billing purposes, while the high-frequency information is for network management. However, the workload of the TTP is high, and the development costs increase since all anonymous IDs are processed here. Moreover, with the introduction of the TTP escrow, the privacy risks are not eliminated but just shifted from the utility to TTP.

A down-sampling method is a naive approach that aims to reduce sensitive information by reducing the interval resolution of the metered data [49, 87, 88]. However, like other methods, demand response and TOU billing functions would be

sacrificed. Moreover, value-add services that require high-resolution data are unavailable as well. To quantify the privacy loss with different interval data, G. Eibl and D. Engel adopt NILM as an adversary to extract personal information. They apply an edge detection NILM to smart meter data and examine the performance of 15 appliances via F-score values and the proportion of appliances. They conclude that 15-minute interval data already protect most appliances.

2.3.1 The knowledge gaps

Existing research points out that there is a strong correlation between the high granular electricity data and the activities/behaviours inside the house [13]; by analysing the real-time power consumption curve, the adversary can determine the personal information such as appliance usage information, presence/absence, and lifecycle [14]. Although privacy concerns are raised in the literature, the threat/adversary model and the data mining algorithm used by the adversary/threat to infer private information are not well defined currently.

There have been various attempts to provide a strong guarantee to the smart meter data, technical solutions in the literature either reshape the actual load curve with rechargeable battery [15] and energy storage system [16] or manipulate the data by distorting [17], aggregating [18], and down sampling [19]. However, it is argued that introducing extra devices will increase the expense of Energy Supplier (ES) and manipulating the original data will influence the billing correctness and make functions, e.g., Time-of-Use (TOU) tariff, unavailable. Furthermore, almost all technical solutions overlook the participation of third-party service providers, who would like to access the smart meter data to provide consumer commercial services. Such value-added services can introduce new market opportunities and engage the innovation of the electricity market [20].

On the other hand, with the steady increase of the distributed renewable generation and electric vehicles, the smart meter data is expected to help the Distribution Network Operator (DNO) improve the visibility and reliability of the Low Voltage (LV)

network. The Office of Gas and Electricity Markets (OFGEM) [21] and the Department for Business, Energy & Industrial Strategy (BEIS) [22] require the DNO to access the aggregated smart meter data inside of individual smart meter for privacy consideration. However, the existing smart metering system does not build the data aggregator to provide such aggregated data. Moreover, how to utilize the aggregated data to improve the visibility and reliability of LV network is not well investigated in the literature.

2.4 Data Privacy Law/Regulation

Current regulation on smart meter data, especially the AI application of the smart meter data, is very limited [89]. This is mainly because private companies are developing AI-based applications quicker than regulators understand their functioning and social implications [90]. In an exploration of documents in the US, the UK and the EU, [91] agree that one main concern is creating or strengthening current regulatory frameworks, like the GDPR, in light of AI's new challenges. They all had different views on how such changes should occur and who is to participate according to the guiding values, though recognising the importance of multilateral and inter-regional discussions.

The timeline of important data privacy laws/regulations in the EU and US is presented in Figure 2-5, referring to the enaction year; these laws are classified into two groups: old guard between 1974 and 2000; and new wave, from 2018 to now. Back in 1974, US Privacy Act was the earliest data privacy regulation. The purpose of the US Privacy Act is to balance the government's needs and the rights of individuals. The private information of individuals is protected from unwarranted invasions. The EU's first data protection directive, Directive 95/46/EC, was enacted in 1995 [92]. Directive 95/46/EC aimed at protecting human rights and freedoms while processing personal data.

Furthermore, Health Insurance Portability and Accountability Act (HIPAA) [93] and Children's Online Privacy Protection Act (COPPA) [94] were enacted by the US in

2000, which are designed to protect the health information and children under 12 years data, respectively. In May 2018, GDPR took effect and replaced Directive 95/46/EC [95]; GDPR proposes more detailed data regulation requirements with a global scope. Like GDPR, California Consumer Privacy Act (CCPA) gives consumers more control over personal information and restricts how companies collect and use data in California [96]. The following will discuss the existing laws and regulations by country in detail.

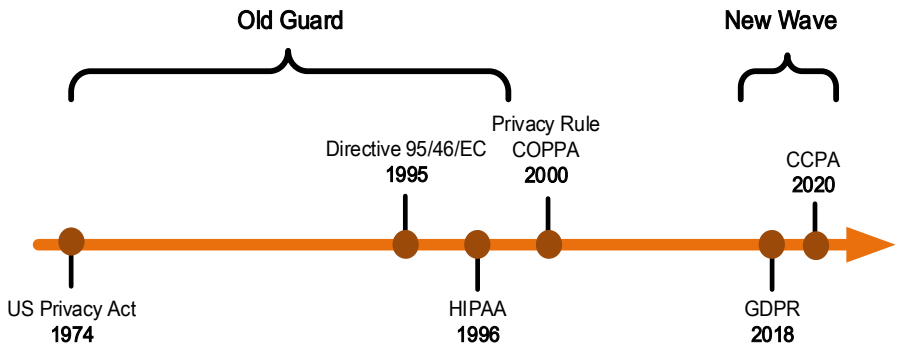


Figure 2-5. Data privacy law/regulation timeline (Adopted from [97]).

2.4.1 Data regulation law in the European Union

By the end of 2020, the total number of smart meters installed in all European Union (EU) Member States will have reached 123 million, increasing the smart meter penetration rate from 34% in 2018 to 43% in 2020 [98]. With the steady increase of the installation number, whether existing data protection regulations and laws can provide excellent protection to smart meter data should be carefully investigated.

GDPR, a replacement of the previous EU Data Protection Directive 95/46/EC (the Directive) [95], became effective on 25 May 2018. As the strictest data protection laws, GDPR guarantees EU citizens' legal rights when data are collected, processed, or shared among different parties [99]. One significant change of GDPR is that it redefines the meaning of personal data, which is now defined as "any information relating to an identified or identifiable natural person". Moreover, location data and online identifiers which can leave traces are also included in personal data. The data protection principles provided by the GDPR are largely based on the Directive, which

includes: (1) The lawfulness, fairness and transparency principle; (2) the Purpose limitation principle; (3) the Data minimization principle; (4) Accuracy principle; (5) Storage limitation principle; (6) Integrity and confidentiality principle [99]. Referring to the description of the GDPR, the power consumption data collected by the smart meter fall within the GDPR. The GDPR should regulate data acquisition, storage, usage, and analysis activities.

More specifically, in 2011, European Commission published a report named "Essential Regulatory Requirements and Recommendations for Data Handling, Data Safety, and Consumer Protection " [100]. This report provides a fundamental basis for regulatory action on the smart grid and smart meters that is to be taken. The report proposed recommendations on privacy and data protection of smart grids and smart meters, and the recommendations contain four parts: business continuity plans, privacy by design and by default, privacy impact assessment, and data retention.

2.4.2 Data regulation law in the US

There has been a long history since the US government published its first data privacy law. The US government also issued laws for certain information, such as HIPAA for health data and COPPA, to protect children's privacy. However, the large amount of data collected by internet companies and the advanced data analytics tools used by these companies (such as Google and Facebook) makes the original data privacy law insufficient to protect consumers' privacy. In this context, in 2020, an extremely powerful, far-reaching law, CCPA, will be signed into law as a state statute to enhance privacy rights and consumer protection for residents of California. Unlike GDPR, which regulates data privacy across the EU, CCPA is only applied to natural persons who are California residents. The CCPA defines personal information as any information that identifies, describes, relates to, or can be associated with a consumer [96]. However, unlike GDPR, which covers all for-profit entities and not-for-profit organisations, CCPA only covers for-profit entities ('business') around the world which sell California residents' personal information. Referring to CCPA, California residents have the authority and the right to decide whether to sell their data to third

parties and the right to request disclosure and delete of the data already collected [101].

2.5 Smart meter ethics

As a category of IoT device, the smart metering system keeps gathering the private data from billions of houses, revealing the personal information about their energy consumption, occupation, lifecycle, inside or outside their home, etc. The great amount of information collected and the advanced data analytics algorithms make the smart metering system outperforms the development of data protection regulation and law [102]. Little research emphasises the ethical and legal implications of utilizing smart meters in the power system industry [103]. Hence, the ethical issue of the smart meter has become a big problem in society. To solve these issues, an ethical framework should be developed to ensure the smart metering system operates morally.

2.5.1 Consumers' worries about smart home devices

'Home', from the point of view of consumers, is the last sanctuary to protect their privacy. The installation of smart sensors such as smart meters and smart appliances is an intrusion into their home, and these devices are not under their control and cannot remove as they want. There are two types of consumers who worry about the smart meter: 1) they put high expectations in smart meters and get disappointed because of the limitation of featured services; 2) they worry about privacy and loss of control of their data. The root of consumers' worries is that they feel they are losing control of their data; they neither be given a suitable insight view about their energy consumption nor be informed when their data is shared with unauthorised third parties.

To settle their worries, the priority is to figure out the system's data flow and the stakeholders involved. [104] proposed to divide the usage of smart meter data into two categories: domestic data flow and remote data flow. Domestic data flow exchange data between smart meter, private platform, and in-home display unit. Data is kept within the home, and any data processing is typically performed in the meter or the

in-home display unit. Domestic data flow enables functions such as real-time power consumption and energy awareness. The domestic data flow has a low privacy leakage risk since no data leave home. Remote data flow shares data with TP or DNO. Although remote data flow can facilitate more sophisticated data analysis, more flexible presentation of information to consumers, and potentially a greater reduction of demand, the risk of privacy leakage also increases.

The different granularity of the data can reveal different information about consumers' privacy. Figure 2-6 summarises the information that can be inferred from data of increasing resolution. Smart meter data with 1 min interval can detect most appliances, while half-hourly data can only infer occupancy information. Most smart meters currently being installed worldwide log data hourly, half-hourly or at 15 min intervals [104]. This can provide a strong indication of occupancy but has much less potential to reveal individual appliance use. Future generations of smart meters may be configured to provide much higher resolution. Concerns should be rationalised accordingly.

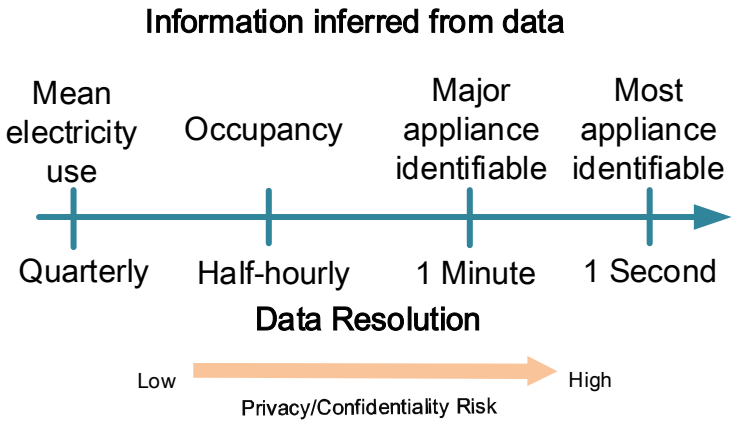


Figure 2-6. Representation of information that can be inferred from metering data in function of the resolution (Adopted from [104]).

Different ways to use the data will also result in different levels of privacy leakages, as demonstrated in Figure 2-7. In [105], the types of personal data are divided into three categories: (1) self-reported data. Data that are reported by consumers voluntarily and contain little sensitive information as consumers already filter out sensitive information. (2) Digital exhaust. The data is collected by smart meters

automatically. (3) Profiling data. This data is evaluated by utilizing advanced analytical tools or algorithms on data collected by smart meters, and the profiling data could reveal consumers' behaviours and interests. The privacy/confidentiality risk of self-reported data is the lowest, and profiling data is the most sensitive data but also has a high value that attracts companies' attention. The usages of the data are also classified into three groups: (1) Improve user experience. Functions such as energy awareness provided by smart meters to improve consumers' satisfaction normally are acknowledged by consumers and make them feel it is a fair business for their data. (2) Facilitating targeted marketing. Consumers' data are collected and used for marketing or advertising purpose. (3) Sell to third parties. This usage has little relation with consumers but is more beneficial to companies. To sum up, the sensitivity is lowest when consumers use the smart meter data to improve their satisfaction, while the sensitivity increases dramatically when their data is sold to third parties to analyse their profiling data. As for the correlation between sensitivity and privacy/confidentiality, it is observed that the more sensitivity to data, the more privacy/confidentiality needs to be protected.

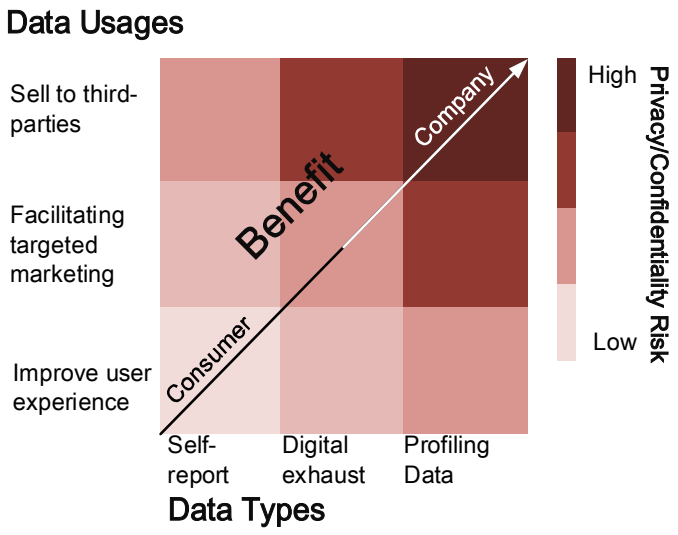


Figure 2-7. Swapping value for data (Adopted from [105]).

2.5.2 Ethical design for the smart metering system

An ethical design for the smart metering system should preserve privacy and provide various ethical choices for the consumers to make them willing to enjoy the added value the smart metering system brings them [102]. Moreover, all ethical designs should be embedded into the hardware and software. In [102], several suggestions for ethical design are proposed:

Privacy by design (PbD): Embedding privacy into the design of the smart meter and the smart metering system proactive is the most efficient way to preserve privacy [102]. Privacy by design means that the data protection is a default smart meter setting without additional add-on sensors or components. Meanwhile, privacy by design employs a 'win-win' mode in which full functions should be retained while privacy and security are guaranteed [106]. Moreover, the smart metering system should be open to maintain visibility and transparency.

Data minimization: The smart metering system should also follow the concept of data minimization, which means only the data needed for certain services is retained to reduce the gathering of personal data [102, 107].

Data anonymization: It is important to remove any information that can show the identity of individuals; hence the third party cannot distinguish the user's identity from the collected database [107].

Control of data: Consumers should be able to manage and control their data during collection and transmission [102].

Ethical options: The ethical design should enable the consumers to be free of their own ethical choices, which contain a different degree of privacy and data protection [103].

2.5.3 Ethical challenges for smart meter

However, there are still several ethical challenges to designing an ethical, smart metering system, referring to the literature. The challenges can be summarized as follow:

- ***Ownership Identification*** [104]***:*** The ownership of the data during data collection and transmission is difficult to identify, and it is a big issue when the smart meter collects personal information without the consumer's consent.
- ***Privacy borderline*** [74, 102]***:*** The data collected by the smart meter could either be sensitive or insensitive; it is essential to define the borderline/boundary between the private and public information; as a result, the consumers can better enjoy the services provided by the smart metering system while their private data is protected. However, it is always difficult to define the borderline.
- ***Life attacks*** [102]***:*** As a smart meter/ smart appliance is installed inside the home, it directly connects all consumer's environments with the internet. Therefore, the smart metring system can directly influence residents' lives by attacking and controlling home energy and further damaging the environment and even people's lives.

2.6 Privacy Design Strategies

GDPR enforces all organizations/companies that process personal data to legally follow the data regulation when collecting, processing, and managing personal data [108]. One of the biggest challenges to implementing GDPR is transferring GDPR obligations into software requirements and designing GDPR-compliant software. However, many software and IT companies do not prepare well for GDPR. PbD is one of the methods to solve the problem stated above, and it is a widely adopted approach to protect private data; this design philosophy guarantees privacy throughout the whole system development lifecycle [109]. A. Cavoukian [110] proposed seven foundational principles of PbD in 2009 and can be summarized as follows: 1)

Proactive, not reactive: PbD involves anticipating events that affect privacy before they occur. 2) Privacy as a default setting: the default setting must be designed to provide maximum privacy protection. 3) Privacy Built-in: Privacy must be integral to systems, applications, and services. 4) Full functionality: PbD seeks an optimal balance for privacy and fully functional solutions. 5) End-to-end security: privacy needs to be assured throughout the data lifecycle. 6) Visibility and transparency: guarantee privacy can be demonstrated and verified. 7) Respect for users' privacy: keeping the user cantered, the rights and freedoms of users must be guaranteed. Nevertheless, PbD lacks clear guidelines to transfer specific legal data protection requirements into system requirements, which limits the application of PbD. To overcome the limitation illustrated above, a privacy design strategy has been proposed, and a design strategy describes a fundamental approach to achieve a certain design goal.

Existing work [108] classifies privacy design strategies into two categories: data-oriented and process-oriented, as shown in Figure 2-8 and Table 2-3. The data-oriented strategies address minimising the privacy impact of the data; data-oriented strategies can be divided into four sub-strategies: Minimise, Sperate, Abstract, and Hide [111]. Moreover, Aggregate is also included sometimes [109]. Data minimization is the most straightforward and obvious strategy; this strategy reduces the privacy risk by excluding, selecting, stripping, or destroying the collection, storing, or operation of personal data. The second strategy, Sperate, highlights that personal data should be processed in distributed or isolated manner whenever possible [108]. Unlike the centralized model, which puts all collected data together, the decentralised or distributed system prevents adversaries from gaining enough information to infer personal data. The abstract strategy focuses on limiting the level of personal data details, and the less detailed a personal data is, the lower the privacy risk it has [111]. The last strategy, Hide, addresses confidentiality, unlinkability and unobservability of personal data; this strategy protects personal data by restricting, obfuscating, dissociating, and mixing the personal data to limit data observability.

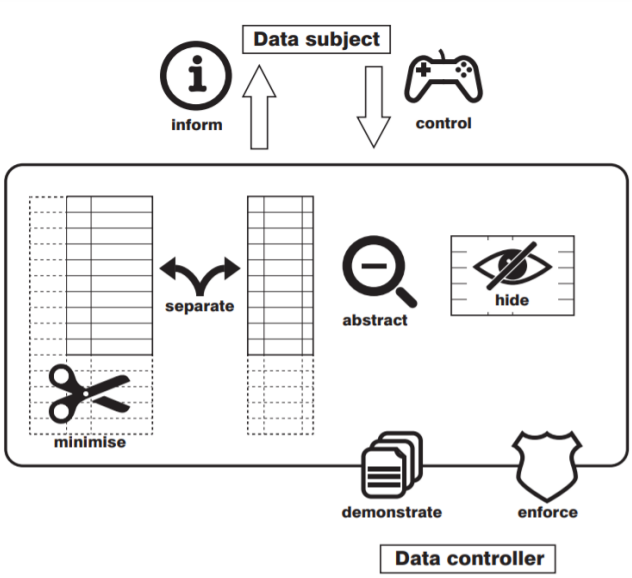


Figure 2-8. Block diagram of the privacy design strategies (Adopted from [112]).

Process-oriented strategies focus on the interaction between the data subject and data controller and address privacy while handling personal data. Process-oriented strategies contain four sub-strategies: Inform, Control, Enforce, and Demonstrate [109]. Inform aims to increase the transparency of processing the personal data, the organizations must supply information about the details of the personal details to be processed and explain the reason for collecting such data; moreover, both the users and society at large should be notified in real-time base to make sure the organizations can be monitored. The second strategy, Control, is one of the fundamental and the most important privacy design strategies; control allows the users involved to handle personal data, users have the right to choose wanted functionalities, and they also have the authority to update or even retract their personal information [113]. Concerning Enforce, the organisations should take responsibility for creating a privacy policy and enforce the policy by using all necessary technical and organisational controls. Finally, the strategy Demonstrate aims to record and audit the system logs of processing data and report the evidence to the Data Protection Authority (DPA).

Table 2-3. Summary of the privacy design strategies (Adopted from [112]).

Data Oriented Strategies		**Process Oriented Strategies**	
Strategy	**Description**	**Strategy**	**Description**
Minimise	Limit the processing of personal data as much as possible.	Inform	Keep informed users about the data process information.
Separate	Separate the processing logically or physically.	Control	Provide data subjects with mechanisms to control the processing of their data.
Abstract	Limit the amount of detail of personal data.	Enforce	Develop the privacy policy for processing data and enforce this.
Hide	Mask personal data to make them unobservable.	Demonstrate	Maintain evidence that you process personal data in a privacy-friendly way.

Although the privacy design strategies provide sufficient guidance for the organizations to design a GDPR-compliant software and make PbD more concrete, there are still limitations to implementing such strategies. Firstly, this strategy only focuses on GDPR and is suitable for organizations in the EU; whether the strategy complies with data regulations in other countries should be investigated. Secondly, since different organizations/companies collect different categories of data (e.g., medical organizations collect clinical data of patients, energy suppliers collect electricity consumption), this privacy design strategy should be modified in specific cases to settle the ethical issues better.

2.7 Advanced Applications with Smart Meter Data

In this section, state-of-the-art applications with smart meter data are introduced. The applications range from appliance-level, household-level, feeder/substation level, to distribution level, as shown in Figure 2-9. The typical appliance-level application is Non-Intrusive Load Monitoring (NILM) which requires high-frequency power consumption data and can disaggregate the load into individual appliances' loads. Appliance-level applications are the biggest threats to individual privacy as they collect personal data and infer individuals' activities from the collected data. As the applications move to higher hierarchical levels such as feeder-level or distribution

level, the sensitivity of the smart meter data is decreased as the data is aggregated, and the individual information will not be recognized.

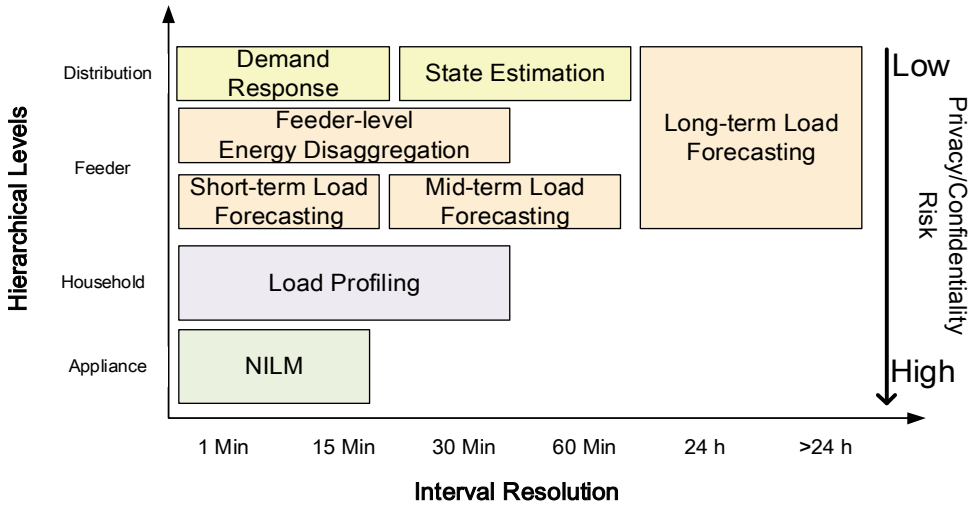


Figure 2-9. Smart meter data application with different hierarchical levels and interval resolutions.

2.7.1 Appliance signatures and Nonintrusive Load Monitoring

Appliance signatures indicate the characteristics of specific appliances, and by identifying the appliance signatures, it is possible to classify different appliances [114]. It is vital to analyse the appliance signatures of household loads and determine how smart meters and AMI use them. Due to the massive application of power electronic devices, most appliances have obvious characteristics in appliance signatures, and Figure 2-10 shows typical current waveforms of household appliances. As for purely resistive loads such as kettle, the steady-state current is sinusoidal and transient current is almost the same as steady-state current; desktop, which represents rectifier loads, has short transient and large peak current (normally several times of steady-state current) in the first few cycles, after that, the system maintains steady-state, and the waveform of steady-state current pulses near voltage peaks; vacuum cleaner, which is a typical motor appliance, due to the accelerating period at start time, it has specific start-up process and decay in transient due to increasing back EMF of motor loads; lastly, the magnetic ballasts in a lamp, which represents inductive loads, since the bimetallic switch operates differently every time depending on the initial

condition, the transient current envelope is also different every time, and the waveform of the steady-state current is sinusoidal and asynchronous with voltage.

As indicated in [115], appliance signatures are classified into two categories: steady-state and transient-state signatures. Steady-state signatures are features extracted from the appliances operating at a steady level of power consumption without transition from another operation state. The typical characteristics of steady-state signatures include:

(1) **Power change:** Power change is the most obvious characteristic which can provide an insight view to observers about the power consumption of appliances. A two-dimensional space of active power P and reactive power Q is introduced to describe the power change [116]; this space decouples the complex power data into two parts and provides a useful graphical interpretation. However, some of the specific signatures would overlap and become difficult to detect [114].

(2) **V-I Features**: V-I features are adopted to deal with the drawbacks of the two-dimensional space; it highlights the characteristics by appending additional voltage and current features into the information (e.g., the Root-Mean-Squared (RMS) values of voltage and current), then the appliance signatures with similar power consumption can be clearly distinguished further.

(3) **Harmonics:** Higher harmonics in the aggregate current signal can be used to distinguish loads with overlapping clusters in the $P - Q$ signature space. Most harmonics are generated due to the presence of power electronics. Particularly, nonlinear appliances such as motors can produce specific harmonic waveforms, which can assist the classification further. A three-dimensional space (P, Q, and Harmonics) of appliances is introduced in [116], which have similar active and reactive power. However, by introducing third harmonics, these two appliances are distinguished easily.

Transient-state signatures represent those appliance signatures of consumption behaviours of appliances transient from one state to another. Transient-state signatures have a close relationship with physical tasks the loads perform, and load monitors

recognize the appliance signatures mainly by specific load transient shapes rather than steady-state signatures.

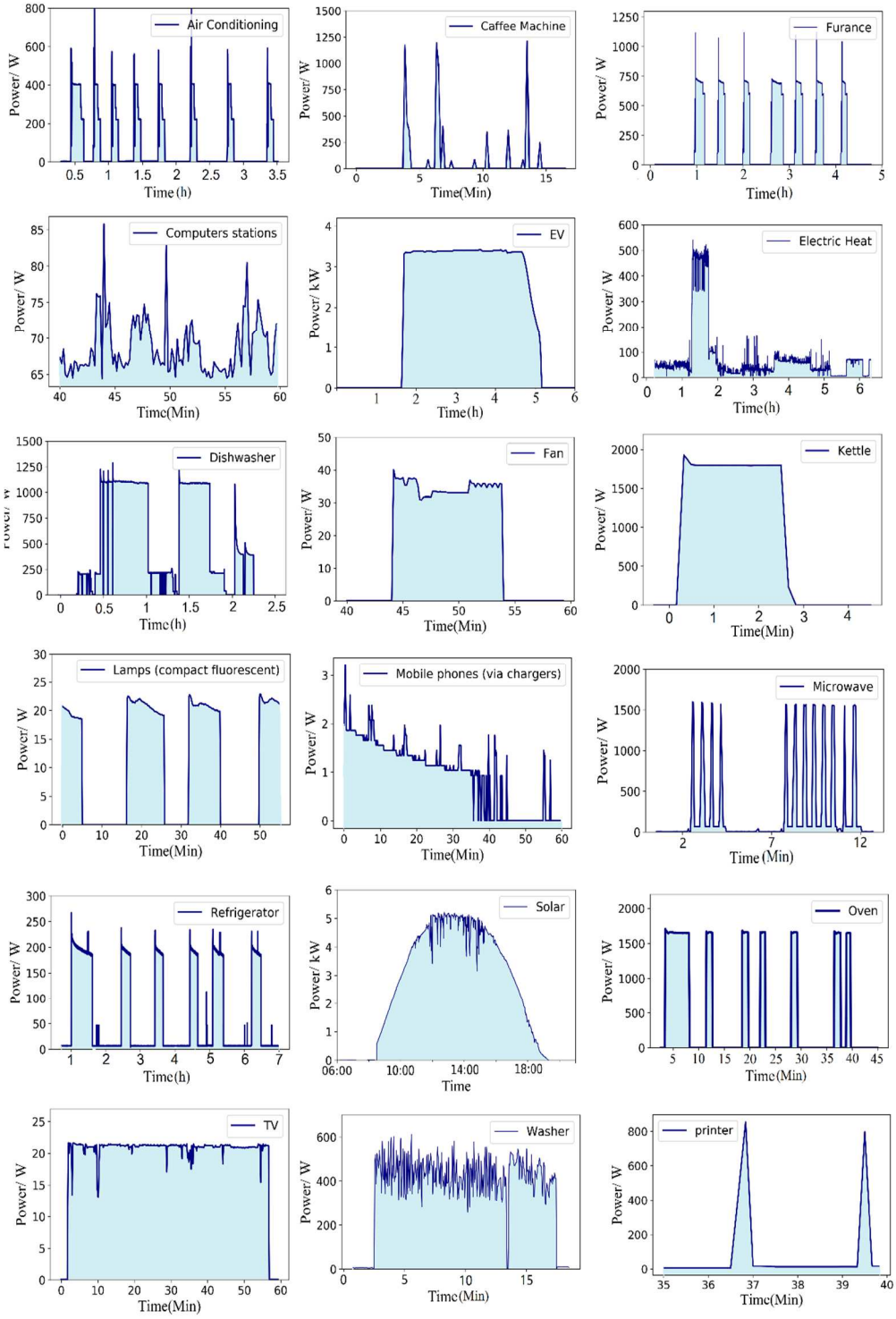


Figure 2-10. Typical appliance signatures (Data source: Pecan Street Dataport [117]).

Proposed by G. Hart in 1992 [115], NILM techniques extract the power consumption of single appliances out of aggregated power data. Liang et al. [118, 119] firstly proposed a four-layer load disaggregation framework by analysing load signatures, and the framework utilizes multi-features (Active and reactive power, harmonics, current waveforms, and et cetera.) to distinguish different electrical appliances. These features can be further used as inputs in disaggregation algorithms. A series of algorithms have been proposed in the last decades, including Deep Learning [120], Hidden Markov Model [121, 122], and Graph Signal Processing [123, 124]. The energy disaggregation technology can be further divided into two categories, which are household-level and substation-level energy disaggregation. By applying the algorithm, the overall power usage data is disaggregated into the real-time power usage of individual appliances. Let $P(t)$ represent the power usage of one household recorded by the smart meter at time t, and the NILM algorithm can be expressed by the following equation:

$$P(t) = \sum_1^n p_i(t)\ (i = 1,2 \cdots n) \tag{2-1}$$

where $p_i(t)$ is the power consumption of individual appliance i (range from 1 to n) at a period t. Moreover, the accuracy of typical NILM algorithms is listed in Table 2-3, and the research has already achieved an extremely high accuracy rate, especially applying LSTM and Bayes models. Although NILM technologies bring advantages to the consumers (energy utilization optimization, energy saving *et al.*), it is still unclear whether the utility of third parties would abuse the data they collected for other purposes rather than consumers required (unauthorized big data analysis to individual consumption data for advertising purpose *et al.*).

Table 2-4. Performance of NILM algorithms.

Algorithm	LSTM	GRU	CNN	KNN
Accuracy (%)	80-97 [125-127]	80-97 [128, 129]	75-98 [130-132]	70-90 [133, 134]

2.7.2 Value-added service platform

Value-added services that utilize fully smart meter data can help the customers better predict, manage and save energy [45] and enhance the customers' living experience.

Typical value-added services include billing, NILM, load forecasting, load management, demand response, customized information services etc. [135, 136]. In the IoT era, the AMI and its surrounding appliances and environment construct the Internet-of-Energy (IoE) [137]. IoE-based platform, which adopts fog computing, cloud computing, and edge computing, has been proposed in the literature.

A centralized cloud-based value-added service platform is introduced in [138-140]. M. Tao et al. [138] develop a multi-layer cloud architectural model which enables interaction between service providers and household appliances; the cloud-enabled platform solves the heterogeneity issues by employing the ontology method. Lloret et al. [139] propose an integrated IoT AMI that can be deployed in smart cities; the centralized architecture relies on a cloud server which utilizes big data/machine learning technologies. The platform enables multiple value-added services, including consumption prediction, incident detection, and customer characterization. In [140], A. Meloni and L. Atzori introduce a virtualization middleware to improve the capabilities and opportunities of the cloud-based value-added service platform. Although cloud computing-based platforms can analyse and process powerfully on the cloud, several shortages should be noticed. First, the scalability of the cloud-based platform is limited; all smart meters in the smart city communicate with the cloud server on a real-time basis; this process generates abundant data to be transmitted, analyzed, processed, and stored simultaneously in the cloud. For a city with a large number of smart meters, the platform would have a high demand of communication bandwidth and cause serve latency which cannot be acceptable for real-time services. Secondly, the cloud-based platform relies on the communication channel and the Internet; all services will be unavailable once the Internet is disconnected. Furthermore, the privacy and security issues raised by the cloud server cannot be avoided as well. Detailed power consumption data of the consumers need to be uploaded to the cloud to obtain certain services; this information could reveal consumers' private information such as behaviour patterns.

Fog-computing and edge-computing, as two distributed computing frameworks, are technologies that overcome the drawbacks of the cloud computing method; these two

approaches enable real-time analytics with little latency. Fog computing introduces an intermediate layer between a cloud server and IoT sensors/devices, and the collected data is processed within a fog node or IoT gateway, which is close to IoT sensors [141]. The first fog computing-based AMI was introduced by Y. Yan and W. Su [142] in 2016; a master node is designed to process and store metadata of a group of smart meters. However, detailed value-added services are not introduced in this scheme. M. Moghaddam [143] and M. Forcan et al. [144] extend the fog-computing scheme, respectively. Services including optimal day-ahead energy consumption schedule, voltage profile monitoring and power loss estimation are simulated in their proposed schemes. The simulation-based study demonstrates that the fog-computing AMI scheme can efficiently reduce total simulation time and data being sent to the Cloud.

Unlike the fog-computing methods, which require extra fog nodes, the edge-computing AMI scheme process the metadata directly on the smart meter. [145] proposes edge analytics-enabled smart home architecture with local deep learning /machine learning configured in the home edge gateway. The architecture can provide energy efficiency services, including home energy management. [146] presents an intelligent edge computing-based energy management for future smart cities. An energy edge server is deployed as the gateway to computing energy data in a local area network. Expect for the power consumption data collected by the smart meter, the edge computing infrastructure's inputs also cover various variables related to the buildings, such as temperature and humidity. However, both works require installing an extra edge home gateway to enable edge computing, increasing the budget for smart meter roll-out. The most relevant work is presented by Matteo Orlando et al. [147], an IoT-enabled 3-phase smart meter designed to enable both on-board, on-fog, and on-cloud services. Moreover, the self-configuration and auto-update procedures help the AMI update its algorithms without affecting the rest systems.

Although edge-computing-enabled AMI has been discussed in the literature, little attention has been paid to the consumers' privacy when providing TPSs to them. Methods such as rechargeable batteries for smart meters [57], noise-adding methods [72], and data anonymization methods [85] have a different level of privacy guarantee,

but these technologies also disable the TPSs as the original smart meter data is distorted or masked. A method which can balance the privacy and functionalities is desired. As a promising privacy-preserving distributed computing technique, federated learning has drawn more and more attention as it allows distributed computing while the clients do not share their data to the cloud [148]. FL has been applied in power system fields such as solar irradiation forecasting [149], electricity consumer characteristic identification [150], and energy management [151]. However, these applications are limited to the interactions between retailers/PV stations and the server; little work emphasizes customer-level applications.

To summarise, the following knowledge gaps in existing literature should be filled:

(1) The flexibility and scalability of value-added services should be taken into consideration for the development of the next generation AMI;
(2) Lacks work to make a trade-off between functionalities and privacy;
(3) Lacks built-in deep learning algorithms to process time-series data efficiently.

2.7.3 Short-term load forecasting

Electric load forecasting can be divided into (1) Short-Term Load Forecasting (STLF), few minutes to 24 hours; (2) Long-Term Load Forecasting (LTLF), one to ten years; and (3) Medium-Term Load Forecast (MTLF), one month to one year [152]. Among all load forecasting methods, STLF Load forecasting is vital for power systems, as it can help power system operators make decisions about supply plans, demand-side management, generation reserve, and so on [153]. In recent years, the prediction performance and reliability of STLF models have improved significantly with the development of AI techniques [154]. Modern STLF models can be divided into single STLF models and hybrid STLF models. Whilst single STLF models can be divided into learning-based models (including regression-based models, deep learning-based models, and machine learning-based models). Conventional regression models include Linear Regression (LR) [155, 156], Gradient Boosting Regression (GBR) [157], ARIMA [158].

Deep learning-based methods utilize multiple hidden layers to evaluate the non-linear correlations between the inputs and outputs of the model, both Convolutional Neural Network (CNN) [159], long short-term memory (LSTM) [160], extreme learning machine (ELM) [161] have been employed to the STLF tasks and achieve good prediction accuracy. Among all DL models, LSTM and its variant Bidirectional LSTM (BLSTM) draw the most attention of researchers for its superior performance in processing sequence data. The memory cell enables Comparing to normal point-to-point predictions, probabilistic STLF methods predict an area of the future load may locate [162, 163], and the probabilistic models can better capture the load variation.

Considering the uncertainty and non-stationary and non-linear properties of load demand, conventional single STLF models are inefficient in extracting the fluctuating loads. Hybrid STLF models have drawn more and more attention in recent years for their high adoption and precise prediction accuracy. Hybrid models usually consist of two or more single methods to better extract the features of inputs and increase the prediction accuracy. Specifically, hybrid deep learning-based approaches that combine the Micro-Clustering (MC) task are introduced in [160, 164-168]. Normally, electric load clustering consists of four steps [154]: Pre-processing, cluster and centroid, construction of the representative load curves, and assess the clustering performance. Whilst traditional MC-based STLF methods [160] cluster the load curve over a time span and ignore the variations of different hours, H. Jahangir et al. [164, 165] proposed an STLF model which combines BLSTM with MC technique smoothly, the load demand data for each hour is clustered into several categories by implementing either supervised or unsupervised MC methods. A specific BLSTM model is trained for each cluster. As a result, the MC-based method can better predict these hours with more spikes [165].

However, these methods discussed above encounter a bottleneck as models only utilize time-domain information, while rich information about the load in the frequency domain is overlooked. The hybrid methods which combine decomposition techniques and DL models can utilize both time-domain and frequency-domain.

Decomposition methods include Empirical Mode Decomposition (EMD) [169], Variational Mode Decomposition (VMD) [170, 171], seasonal and trend decomposition using Loess decomposition (STL decomposition) [172], and Empirical Wavelet Transforms (EWT) [173]. EMD-based STLF methods are introduced in [169]. As an adaptive non-linear decomposition method, EMD decomposes the original signal into a series of Intrinsic Mode Functions (IMFs) using Hilbert–Huang transform, and each IMF is an amplitude modulation–frequency modulation (AM-FM) signal [174]. However, as a purely data-driven method, EMD lacks a mathematical definition, so it is difficult to understand the decomposition results; secondly, the decomposed signals will diverge at the endpoints and highly sensitive to noise [12]. VMD-based STLF methods are presented in [171, 175]. As an alternative algorithm to EMD, VMD is a non-recursive, adaptive decomposition estimation method to decompose the original signal into several mode functions with specific bandwidth in the frequency domain [176]. In [171], S. Kim et al. introduced a hybrid STL-VMD-LSTM STLF method to extract both seasonal and frequency features of the electric load. K. Semero proposed a hybrid VMD-ANFIS forecasting model [175]; the model takes advantage of both mode decomposition and fuzzy logic principles. The latest decomposition algorithm, EWT, combines the strength of the wavelet's mathematical definition with the flexibility of EMD [173].

Although there are a rich of works have been illustrated in the literature, the existing STLF models still have some knowledge gaps that can be filled:

(1) Firstly, the hybrid DL with mode decomposition methods in the literature either lack mathematical definition (EMD) or low adaptivity (VMD); a new hybrid STLF which takes advantage of both EMD and VMD should be proposed.
(2) Secondly, electric spikes and other noise would influence the training process and the prediction accuracy; a proper denoising technique should be selected to process the original data.

2.7.4 Solar energy separation at the grid supply point

Renewable energy generation capacity, such as solar panels and wind turbines, is growing rapidly to reduce the emission of carbon dioxide and the costs of energy consumers. However, most renewable generation is installed Behind-The-Meter (BTM), which means the generated solar energy is invisible to the energy utilities. These BTM PV systems have significantly changed the netload's shape (netload in this thesis is defined as the customer's energy consumption minus its solar generation), which challenges the network operators to determine adequate operating reserves and make precise short-term load forecasting [177]. Hence, a method which can separate the PV generation from the netload is essential for the grid operator to control the distribution network. Depending on the grid level, the solar energy separation methods are divided into customer-level methods, distribution/feeder, and transmission level methods. The consumer-level solar energy separation method is introduced in [178, 179] as a method similar to the NILM technique; the purpose of household-level solar energy separation is to divide the consumer's PV generation from the induvial smart meter data. However, analysing household-level solar energy generation will introduce privacy risks to the customers, which is opposite to the target of this thesis.

The distribution/feeder level solar energy separation method estimates the overall PV generation in a community or an area without intruding individual's smart meter. In the literature, distribution/feeder level solar energy separation methods can be divided into two categories, which are deterministic disaggregation and probabilistic disaggregation methods. Existing work mostly focuses on the deterministic disaggregation method, which can be further divided into model-based, upscaling, and data-driven methods. The model-based method estimates the total solar energy in a region via constructing an equivalent PV system [180-185]. Taking grid measurements, weather information, and irradiance information as inputs, the parameters of the equivalent system are optimized by solving an optimization problem formulation. Few regression methods are proposed, including Contextually Supervised Source Separation (CSSS) methodology [184] and its extension [185],

convex optimization [180, 182], and multiple support vector regression models [183]. In addition, ambient temperature is adopted as a correction factor to improve the model accuracy in some works [182, 185]. In [183], K. Li et al. proposed an Ensemble model by considering sub-models under different weather conditions; the capacity of the PV systems is also estimated automatically.

To sum up, the model-based method requires knowledge of the PV module model (angle of solar radiation, series resistance, etc.); lacking vital information would lead to a large error between ground truth and estimation. Hence the flexibility of the model-based method is limited. Moreover, the mathematical equations have normally highly non-linear and chaotic characteristics, which is computationally intensive.

Upscaling method [186-189] selects a small number of representing PV systems as the reference to estimate the overall PV generation of the whole area. In [186, 187], a method that combines a data dimension reduction procedure and a mapping function (linear regression, Kalman filter) is proposed. Moreover, a hybrid method that combines upscaling with satellite-derived is proposed in [188, 189]. Next-generation high-resolution satellites produce high accurate irradiance estimates that can improve performance. However, upscaling methods require measurements from a small group of PV systems in the target area, and this information is not always available for rooftop PV.

To overcome the shortages of approaches mentioned above, the data-driven method utilizes high-resolution data from a variety of data resources that correlate to solar energy to estimate solar energy [186, 187, 189-192]. The power data comes from micro-phasor measurement units (μPMUs), supervisory control, and SCADA ,and smart meters. While meteorological data comes from National Centres for Environmental Information (NCEI) (US) [193], Climatological Observers Link (COL) (UK) [194], and satellite data from The National Solar Radiation Database (NSRDB) (US) [195], or RE Data Explorer (Central Asia) [196]. F. Bu et al. [190] proposed a game-theoretic approach, a closed-loop game-theoretic approach is used to search the optimal composite exemplars from a candidate library, and a semi-

supervised source separator is employed for disaggregation. In [191], Y. Wang et al. proposed a hybrid method that combines model-based and data-driven load/PV forecasting techniques. The PV generation and demand load are decoupled at first, and each component is forecasted individually via a neural network and Gradient Boosting Regression Tree (GBRT). Moreover, a Multi-Layer Perceptron (MLP) neural network is raised in [192], and measurement data from various sources are fed into the network to implement the disaggregation. The results show that the hybrid model achieves better performance than models that only utilize one data source.

Probabilistic solar energy separation/forecasting methods are developed based on the conventional data-driven approaches introduced above. Compared to deterministic disaggregation methods, as mentioned above, the probabilistic disaggregation method provides more instructive information, such as the quantification of the uncertainty. The network operators can benefit from the probabilistic results by considering the uncertainties to determine necessary reserve deployment and improve the operation efficiency. Probabilistic methods have been employed in solar energy forecasting with great success before, and models such as quantile regression [197], Gaussian process [198], ensemble learning [199], and Bayesian deep learning [200] are introduced in the literature. Few works focus on probabilistic PV separation as there are more uncertainties to disaggregating the PV generations from the netload, as both the PV capacity and historical PV generation data are unknown to the researchers. The authors in [201] introduce a probabilistic PV estimation method to estimate the PV capacity given time-series historical load and irradiance data while considering the uncertainty in solar irradiance and the measured netload. Empirical probability density functions are used to determine the uncertainty, while the Monte-Carlo method is used to simulate the customer load and generated solar energy, and a quantile analysis approach is utilized in the final step to estimate the PV capacity. Moreover, considering the netload data is stored in several communities/energy suppliers, sharing/collecting data will introduce privacy risk. a new approach which combines probabilistic solar energy disaggregation methods with FL is proposed by J. Lin et. al [202] and X. Zhang et. al [203] in 2021 to protect the privacy of the consumers/entities

during the data analytics and data sharing, FL is a privacy-preserving edge-computing approach which does not need personal share data to the central cloud.

To sum up, method-based methods have a strong mathematical explanation of the PV systems but lacking adaptation and flexibility are strictly limited. While upscaling is suitable for large-scale PV systems, data-driven methods heavily rely on data resources and do not have strong mathematical definitions. From the literature, the followings knowledge gaps should be filled in this thesis:

(1) In the literature, most methods are supervised learning approaches, while in most cases, ground truth PV generation data is unenviable for training the model. Hence, a hybrid solar energy separation method which employs either an unsupervised learning model or a supervised learning model should be proposed to increase the flexibility of the application in practice.
(2) A part of research in the literature still requires the acquisition of the data from the individual smart meters, which brings additional privacy risks to the consumers. Hence, the proposed method should avoid accessing an individual's energy consumption or accessing the aggregated data from the aggregators.
(3) Some methods in the literature introduce complex mathematical and physical models, significantly increasing the computation complexity; the proposed method should require lower computation capacity.
(4) The solar energy separation method should have high adaptivity to be applied to the feeders with different PV penetration rates.
(5) The transferability of the solar separation method must be investigated to find the possibility of transferring the pre-trained model to other areas.

2.7.5 Feeder-level energy disaggregation

Feeder-level energy disaggregation technology Artificial Intelligence (AI) based feeder-level energy disaggregation approaches are introduced in [186, 187, 204-211]; the objective of this method is to decouple the feeder-level net demand into load

components. The above approaches can be further divided into model-based and measurement-based methods.

The model-based method is presented in [209-211], which combines the ZIP load model with artificial neural network algorithms. Synthetic data is built based on the ZIP/exponential load model, and then Monte Carlo simulation is used to generate synthetic training and validation data. By changing the weights of load components and voltage, a few active power and reactive power measurements are obtained, which are used for model training/validation. Moreover, a two-layer feedforward shallow neural network is built to estimate the portion of each load category from the total load demand measured at the substation level [210]. A multi-modal LSTM is introduced in [209] to identify the time-varying ZIP load and Induction Motor (IM) model parameters. The algorithm's accuracy is increased by considering different modalities of the input data. The advantage of this method is that the dataset can be easily constructed, referring to the ZIP/ exponential load model. The limitation of this approach is that the dataset used in the case study is synthetic, and the trained model cannot be used in a real-world case.

Unlike the model-based method that uses a synthetic dataset, the measurement-based approach utilizes real-world smart meter measurements. Ledva et al. [205] proposed an online learning method to separate air conditioners' demand (AC demand) from the demand load. Household-level smart meter measurements provided by the Pecan Street Dataport [117] are aggregated to build a feeder-level load. Then an online learning algorithm, Dynamic Fixed Share (DFS), is adopted to perform energy disaggregation by considering measurements from both substation and weather stations. Based on [205], an improved algorithm that combines model-based method and measurement-based method is presented in [206]. Substation, feeder, and smart meter measurements (active power, reactive power, complex voltage, and complex current) are utilized together to enhance the algorithm's flexibility. The online learning algorithm, Dynamic Mirror Descent (DMD), keeps iterating for measurement- and model-based updates. The difficulty of the measurement-based approach exists in the difficulty of obtaining data to train the model.

Researchers further work on increasing the visibility of Behind-The-Meter (BTM) solar energy by decoupling the solar energy from the net load [204, 208]. Unlike traditional demand load, solar energy generation is highly related to solar irradiance data and meteorological data. In [208], a regional NILM algorithm is proposed to disaggregate solar energy and Electric Vehicles (EVs) loads from the substation demand. The data used for the case study combines three data sources (substation demand dataset, solar energy dataset, and EV dataset), where each component is separated individually using their proposed three-stage disaggregation framework. The substation demand is the first forecast via EMD; the solar energy is estimated by matching the linear correlation between the solar irradiance and the PV outputs. Finally, the EVs load is estimated via the Limited Activation Matching Pursuit (LAMP) method. [204] views the energy disaggregation as a partially labelled dictionary learning problem. By training the offline model with historical datasets with partial labels, the system can efficiently separate three load categories, including solar energy. However, in practical application, the situation is more complex than the case study they implemented. There are more than three categories of load aggregated at the same time. Other solar energy disaggregation methods include linear regression, Kalman filter [186, 187], Gradient Boosting Regression Tree (GBRT) [191], Multi-Layer Perceptron (MLP) neural network [192], Gaussian Mixture Modelling (GMM) [212]. Nonetheless, these approaches only focus on separating solar energy, and other load components remain unseparated from their research.

Another research area that correlates to the proposed method is the probabilistic estimation task. Probabilistic estimation was used in power systems and energy discipline with great success, e.g. Load forecasting [213-217], locational marginal prices forecasting [218], Probabilistic Real-Time Thermal Rating (RTTR) forecasting [219, 220], and wind forecasting tasks [221]. Probabilistic estimation utilizes a variety of approaches such as Quantile Regression (QR), Quantile GBRT (Q-GBRT), Regression Neural Network (QRNN) [222], Probabilistic Intuitionistic Fuzzy Time Series Forecasting (PIFTSF) [223] methods to estimate the results in the forms of quantiles Prediction Intervals (PIs), etc. As a typical uncertainty quantification

approach, PIs set the upper and the lower bounds to quantify the level of uncertainty, and the corresponding PI nominal confidence (PINC) is provided as well (for instance, PINC equals 95% with 0.975 as the upper bound and 0.025 as lower bound).

Most relevant to this work, J. Ponoćko and J. V. Milanović [224] present an ANN-based energy disaggregation method; this method combines the advantages of model-based and measurement-based methods introduced above. Instead of synthetic training data, this method utilizes aggregated smart meter data with detailed load composition information. The simulation results show their method can disaggregate the netload into controllable and uncontrollable loads with high accuracy. Although this cutting-edge work provides a more practical scheme, several improvements can still be made based on their work: Firstly, as a point prediction model, ANN has limited learning ability to process sequence data, and RNN models could have better model performance. Secondly, it has been proven that weather and calendar features can boost prediction accuracy [225]. Hence the prediction model should access more external databases. Moreover, the increasing penetration of renewable energy changes the load shape. Hence these distributed generations should also be taken into consideration.

In the literature for feeder-level energy disaggregation, some knowledge gaps should be filled and can be concluded as follows.

1) In the literature, only grid measurements (e.g., active/ reactive power, voltage) are utilized as model inputs. However, load demand is influenced by other variables such as meteorological and calendar data. Hence, a model which considers all relevant variables should be proposed.
2) Although both machine learning and deep learning algorithms are introduced in the literature, the uncertainty of energy disaggregation is not discussed.
3) The transferability of the energy disaggregation technique is not investigated; it is vital to validate whether the proposed method can be used in different areas.

2.7.6 Comparison among three applications

A comparison of the three applications discussed above is presented in Table 2-5. These applications have some similarities, as they all utilize electricity load demand data to solve problems. The purpose of NILM and feeder-level energy disaggregation aims to separate the load into sub-components on a real-time base to increase the visibility of the load. The difference between these two techniques is the hierarchical level; while NILM focuses on a single house or building, the feeder-level energy disaggregation tries to understand the load compositions of a wholly residential area. Hence, the feeder-level energy disaggregation can achieve similar results while it does not invade personal data. As for STLF, the target of STLF is to forecast the overall demand load in the short future rather than the current timestep. The purpose of STLF is to increase the predictability, and the load components are unknown to the power system operator.

Table 2-5. Comparison between three problems.

	Feeder-level Energy Disaggregation	**House-level Energy Disaggregation (Nonintrusive Load Monitoring)**	**Feeder-level Load Forecasting**
Input	Netload at the grid supply point	Power consumption of a single house	Historical demand
Output	Portion of appliance	Load compositions	Future demand load
Aggregation Level	A residential area	Single house or building	A residential area
Horizon	Real-time	Real-time	Future
Privacy Issue	No	Yes	No

2.8 Chapter Summary

In this chapter, a comprehensive literature review is presented. The review focuses on six aspects that strongly correlate with the research topics. The infrastructure of smart grids and advanced smart metering systems was initially introduced. Then the privacy intrusion issues of smart meters are classified into four categories: behaviour patterns identification, real-time surveillance, fraud, and non-grid commercial uses of data. Investigation state-of-the-art privacy enhancement techniques are implemented in Section 2.3, and existing methods are divided into two main groups: user demand

shaping and data manipulation; both the advantages and disadvantages of each method are fully discussed.

Moreover, the latest data regulation policies to regulate smart meter data in Europe are researched, and the correlation between the GDPR and smart meter data is emphasised. Furthermore, the ethical issues raised by smart meters and artificial intelligence algorithms are highlighted in the review, and a ‘soft’ ethical strategy is proposed to settle raised ethical issues in the following chapters. Finally, the advanced applications with smart meter data are listed. To sum up, this chapter provides rich background research and contributes to the rest of the thesis.

Chapter 3 Smart Meter Data Analytics Methodology

This chapter introduces the machine learning/deep learning and data analytics methodologies adopted in this thesis. With the rapid development of the new generation, smart metering systems which enable cloud computing and real-time communication, how to analyse and utilize the data ethically collected by the smart meter is the problem to be emphasized. Moreover, new challenges and difficulties: high data dimensionality, large data volume, and high data acquisition/ transmission speed [226], need to be overcome when researchers monitor or analysis the smart grid. Hence, the Artificial Intelligence (AI)-based data analytics method is an important approach to understanding the characteristics of the smart meter and further allocates significant features associated with the customers' privacy. Energy suppliers, retailers, distributed system operators, and service providers rely on data analytics to support various services and functionalities [227]. Artificial intelligence has been widely employed in power system analytics such as load analysis (including load profiling [228], energy theft detection [229]), load forecasting [216], load management (consumer characterization [230], demand response implementation [231, 232]).

The rest of this chapter will introduce feature engineering and pre-processing, which help the machine learning algorithms select the most relevant variables to make a precise prediction and transform raw data into an understandable format for the machine learning/ deep learning models. Then the fundamental theory of typical machine learning/ deep learning algorithms is demonstrated. Finally, the information theory is introduced to pave the way for measuring privacy leakage in the following chapters.

3.1 The Importance of Smart Meter Data Analytics and The Challenges in The Era of Big Data

As introduced in Chapter 2, the smart grid is a revolution of both Information And Communication Technologies (ICTs) and physical facilities. On the one hand, ICT equipment such as advanced metering infrastructure enables two-way communication of high granular data between the customers and the energy suppliers; on the other hand, the smart grid also enables two-way power flow with the introduction of distributed renewable generation (such as solar energy, wind turbine. The complex infrastructure and the big data collected by the advanced smart sensors brings great potential value for optimizing power system and provide various business opportunities for energy and internet companies.

However, the redundant data also significantly pressures the existing power analytics system and data storage facilities. The big data challenges in the smart grid can be summarized as the four V's referring to [233]: volume, velocity, variety, and value. Volume indicates the large volume of consumer data collected by millions of smart meters in near real-time. The overall volume of 100 million smart meters with a sampling frequency of 15 min reaches 2920 TB every year [226], which Velocity means the big data in the smart grid must be processed at a very high speed to ensure real-time management and operations to guarantee the supply-demand balance as well as quick recovery from the system failures. Variety represents various categories of smart grid big data available for use. In the past, only the power consumption data in kW collected by the conventional electricity meter was utilized.

Nevertheless, in the smart grid, not only rich power system information (active power, reactive power, voltage, current, etc. [234]) is measured, but a large amount of external data such as meteorological information and geographic information is also integrated into the smart grid big data [226]. The value represents big data analytics's huge value to the power system industries. Analysing the big data help power system operators with power generation, transmission, delivery, fault diagnosis, etc. Moreover, the

advanced data analytics approaches also create new business models to provide multiple value-added services to the customers [233].

Another challenge of the smart grid in the era of big data is privacy and security [45]. As introduced in Chapter 2, the privacy issues raised by the smart meter data are the main barrier to data analytics; malicious attackers may also use data mining and artificial intelligence methods to infer sensitive information from the smart meters [235, 236]. Analysing the smart meter data in an ethical and privacy-preserving manner is vital for the smart metering system design in the rest of this thesis.

3.2 Smart Meter Feature Engineering and Data Pre-Processing

Most models are based on the hypothesis that the dataset to be trained is clean and noise-free; any disturbance in the data would influence the performance of the models. However, problems such as missing values, outliers, noise, duplication, and inconsistency cannot be avoided during the collection process by the smart meter [237]. Hence, data pre-processing is the essential and primary step before feeding the data to machine learning/ deep learning models, and it aims to prepare a good quality, reliable and suitable data source for further mining algorithms [238]. In addition, the original dataset may contain too many features, while only a small part of these features is correlated to the output. In other words, these irrelevant and redundant features increase the data dimensionality and decrease the computation speed of the algorithms significantly. Feature engineering tries to remove these redundant features as much as possible to improve the computation efficiency [239].

3.2.1 Data cleaning

Good data quality will lead to better model performance. However, there are many missing values, wrong labels, and duplicates during the data collection. Hence, data cleaning is one of the most important preprocessing steps to add/ remove incorrect,

duplicate, or missing data. Normally, the data cleaning process is divided into four steps (see Figure 3-1):

(1) De-duplicate/remove irrelevant observations. Duplicates always happen during data collection when the data manager merges the datasets from different places and users. Moreover, a dataset may contain hundreds or thousands of different features, while most features are irrelevant to the objective. Hence, removing the duplicates and irrelevant observations will make the data analytics more efficient.
(2) Fix structural errors. Structure errors are these typos or inconsistent capitalization which occur during the data transferring and measuring. These errors could result in mislabelled categories and must be fixed.
(3) Filter outliers. If some values inside a dataset are not fitted within the dataset and are irrelevant to other measurements, these could be outliers measured by mistakes. Once the researchers have enough reasons, these outliers can be removed legitimately.
(4) Handle missing data. Most machine learning algorithms cannot operate with missing values, simply ignoring these missing values will make the dataset lose a large amount of information. There are different approaches to dealing with missing categorical data and missing numeric data. The best method for missing categorical data is to have a new class called “missing” so that machine learning will be informed which data is missing automatically, and it will find the correlations among all missing values; as for missing numeric data, the flag and fill method are adopted to flag these missing values with indicators and make the algorithm estimate the best value for the missing observations.

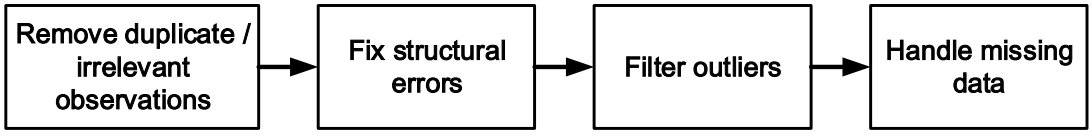


Figure 3-1. Data cleaning process.

3.2.2 Categorical feature encoding

Data are divided into numerical and categorical data: continuous and discrete data are generally numeric as the categories of the data are infinite (such as active power, voltage, current, and temperature); categorical data is non-numeric and can be classified into several groups (for instance zip codes, weather conditions, etc.). In feature engineering, the categorical data must be encoded before sending it to machine learning models, most machine learning models can only recognize numbers, and the algorithms cannot operate normally with non-numeric data. Two standard categorical feature encoding methods: label encoding, and one-hot encoding, are introduced as follows.

3.2.2.1 Label encoding

Label encoding is the simplest method, and it simply converts each category to a certain number (e.g., 0, 1, 2) [240]. However, the number sequence is a huge issue for label encoding, and the machine learning model could give the category with a larger number higher precedence over those with lower numbers. As a result, the model will assign more weights to the category with a larger number.

3.2.2.2 One-hot encoding

One-hot encoding is a binary style of categorizing method that avoids algorithm misinterpretation [240]. In one-hot encoding, each category value is converted to a new column, and then value one is assigned to these columns when this category presents in the current row, or value 0 is assigned when this category is absent. In short, one-hot encoding generates a binary vector with a length equal to the number of categories. However, one-hot encoding produces extremely high-dimensional vector representations, resulting in significant issues in computation memory and computability.

3.2.2.3 Feature scaling

In machine learning or deep learning algorithms, the input data contains multiple independent features on very different scales [241]. Without a proper scaling method, the features with a large scale will impact the output of the model greater than others. Meanwhile, the machine learning algorithms will not perform well under such conditions. Hence, it is vital to pre-process the input data to normalize the range of these different features [242]. Feature scaling is a technique to normalize the range of different data features during the pre-processing data period. Two of the most common feature scaling methods are max-min normalization and standardization.

3.2.2.4 Max-min normalization

Max-min normalisation (max-min scaling) is a normalization method that rescales and shifts the input data to values between 0 and 1 [243]. The equation of max-min normalisation is shown in Equation (3-1):

$$x_{normalized} = \frac{x - \min\ (x)}{\max(x) - \min\ (x)} \tag{3-1}$$

where max (x) and min (x) represent the maximum and minimum values of the dataset.

3.2.2.5 Standardization

Unlike max-min normalisation limit the values into a specific range, standardization makes the new values of each feature have zero mean and unit variance:

$$x' = \frac{x - \mu}{\sigma} \tag{3-2}$$

where μ is the mean value, and σ is the standard deviation of the feature vector.

3.2.3 Feature selection

As stated in the previous section, most of the features in a dataset are redundant and irrelevant to the output values, and feature selection is the method to remove these

redundant features as much as possible while the important information is not lost meanwhile [239]. There are many benefits to implementing feature selection in machine learning algorithms: Firstly, the model is simplified to reduce the server's computation overhead and storage capacity; secondly, less computation time is required. This advantage is significant when the training data size is large; thirdly, the curse of dimensionality problem is avoided. When there are too many features, dimensionality increases, and the space volume increases, making the available data sparse. In such a high dimensional space, the data for training grows exponentially to achieve good prediction results [244]. All features can be classified into three classes: relevant features, irrelevant features, and redundant features. Relevant features indicate these features which significantly impact the output and cannot be replaced by others. Irrelevant features represent the features that do not influence the output. Redundant features are the features that other existing features can replace.

A feature selection approaches contain two components, a selection algorithm to generate candidate subsets and an evaluation algorithm to score the generated subsets. Typical feature selection methods include the filter, wrapper, and hybrid methods [245]. The filter method ranks the variables referring to the relevant scores and uses a threshold to filter out those variables below the threshold, regardless of the model's algorithm. The relevance of the features to the output is measured by ranking methods such as Kullback-Leibler divergence (K-L divergence), mutual information (MI), correlation analysis, and Fisher’s score. Whereas the wrapper method will first generate the subset of features, then the model is trained with the subset; by adding/subtracting features and training the model, the best features to achieve the highest accuracy are obtained. The hybrid method combines the filter method and wrapper method, which takes care of the machine training iterative process while maintaining the computation cost to be minimum. Followings are some typical ranking methods used in feature engineering.

3.2.3.1 Ranking method

3.2.3.1.1 Kullback-Leibler divergence

K-L Divergence is an information-theoretic quantity to assess the similarity between two distributions, and it has diverse applications such as applied statistics, fluid mechanics, neuroscience, and machine learning. The K-L Divergence is defined as:

$$D(P||Q) = \int_{x_{min}}^{x_{max}} f_P(x) log \frac{f_P(x)}{f_Q(x)} dx \tag{3-3}$$

where $f_P(x)$ and $f_Q(x)$ represent the probability density functions (pdfs) of P and Q. The larger value $D(P||Q)$ is, the better privacy has been protected.

3.2.3.1.2 Mutual information

MI $I(X^n; Y^n)$ measures the dependence between two random variable sequences X^n and Y^n [246]. In other words, MI can explain the reduction of the original load sequence X^n given knowledge of the modified sequence Y^n:

$$\begin{aligned} I(X^n; Y^n) &= H(X^n) - H(X^n|Y^n) \\ &= H(X^n) + H(Y^n) - H(X^n, Y^n) \\ &\approx -\frac{1}{n} logp(Y^n) - \frac{1}{n} logp(X^n) + \frac{1}{n} logp(X^n, Y^n) \end{aligned} \tag{3-4}$$

where $H(X^n)$ and $H(Y^n)$ are the marginal entropies, which measure the uncertainty about the random variable; $H(X^n|Y^n)$ is the conditional entropies, and (X^n, Y^n) is the joint entropy of $H(X^n)$ and $H(Y^n)$. This thesis adopts a variant MI named Normalized Mutual Information (NMI) to show the normalized results between 0 and 1 (0 represents no mutual information, 1 represents perfect correlation).

3.2.3.1.3 Correlation analysis

The Pearson correlation coefficient ρ is used to measure whether two continuous variables are linearly associated. The value of ρ ranges from -1 to 1 (a positive value indicates positive correlation, while a negative value indicates negative correlation); the larger ρ, the stronger the correlation between two variables. The expression of the Pearson correlation coefficient is shown in (3-5):

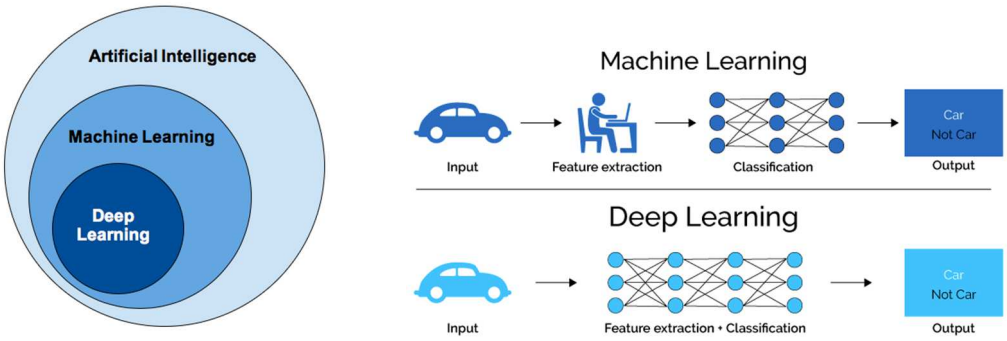
$$\rho = \frac{\sum_{t=1}^{n}(x_t-\bar{x})(y_t-\bar{y})}{\sqrt{\sum_{t=1}^{n}(x_t-\bar{x})^2 \sum_{t=1}^{n}(y_t-\bar{y})^2}} \tag{3-5}$$

where n is the sample size, x_t is appliance power consumption at time t and y_t power consumption generated by the adversary; $\bar{x}, \bar{y}$ is the mean value of x_t and y_t.

3.3 Machine Learning Foundations

AI is a branch of science that develops intelligent computers, systems, or machines to handle tasks that a human should do before [247]. As AI is a broad definition in multiple disciplines such as computer science, psychology, philosophy, and linguistics, a narrow definition should be provided in the field of computer algorithms. Machine learning is the core and fundamental concept of AI, which enables machines to have the ability to learn without being explicitly programmed [248]. As a part of AI, the machine learning algorithm enables machines to perform tasks by gaining experience from past data without programming explicitly [241]. Machine learning can be used for complex tasks for traditional algorithms such as spam filter, image classification, natural language processing, time series forecasting, etc. Different machine learning algorithms can be classified into supervised, unsupervised, semi-supervised, and reinforcement learning, referring to the supervision methods when these models are trained. However, the disadvantages of traditional machine learning algorithms are also obvious (see Figure 3-2): Firstly, traditional machine learning algorithms require extracting features manually, failed to select the right features would reduce the prediction performance; secondly, the structure of many traditional

machine learning algorithms are simple (such as ordinary least squares regression, linear regression), the accuracy of these models decreases significantly when the models deal with a large amount of data. Moreover, conventional machine learning algorithms require long computation time and considerable computation power in many cases, which limits the application in more complex tasks.



(a) (b)

Figure 3-2. (a) Relations among artificial intelligence, machine learning, and deep learning (Adopted from [249]); (b) Difference between deep learning and machine learning (Adopted from [250]).

3.3.1 Supervised learning

Supervised learning is the Machine Learning task that trains a model from labelled data containing training examples. Moreover, each input-output pair of training examples contains an input vector and the desired solution to the input (labels) [251]. The purpose of the supervised learning algorithm is to learn the inferred function that can best map the inputs and the outputs of the training examples, and the function can be used for mapping new examples [252]. Referring to different tasks, supervised learning can be further divided into classification and regression tasks. The target of the classification task is to classify the examples into different categories (face classification, spam filter etc.), while the regression task aims to predict a target numeric value given input (load forecasting, energy disaggregation, et cetera). Some important supervised learning algorithms include k-Nearest Neighbours (KNN), decision trees and random forests [243].

3.3.1.1 K-nearest neighbours

KNN is one of the simplest machine learning algorithms that can be used for classification and regression tasks, and it is a category of lazy learning algorithms that assign an object to the most frequently occurring class amongst its nearest neighbours. K represents the number of data points to be considered in this classification. The distance between every two points is measured by a distance measure function such as Euclidean Distance, Manhattan Distance and Minkowski Distance. Assume two tuples, $X_i = (x_1^i, x_2^i, \cdots, x_k^i)$, and $X_i = (x_k^1, x_k^2, \cdots, x_k^N)$, then the formulas of the three distance function can be expressed as follow:

Euclidean Distance:

$$Dist_E(X^i, X^j) = \sqrt{\sum_{k=1}^{N} \left(X_k^i - X_k^j\right)^2} \tag{3-6}$$

Manhattan Distance:

$$Dist_{Man}(X^i, X^j) = \sum_{k=1}^{N} \left|X_k^i - X_k^j\right| \tag{3-7}$$

Minkowski Distance:

$$Dist_{Min}(X^i, X^j) = \left(\sum_{k=1}^{N} \left(\left|X_k^i - X_k^j\right|\right)^q\right)^{1/q} \tag{3-8}$$

3.3.1.2 Decision tree and random forests

A random forest is an ensemble of decision trees, and the model is trained via bagging generally [253]. As the fundamental component of the random forest, a decision tree is a tree-like model that can be used to present decisions and decision-making visually (see Figure 3-3) [254]. It contains two types of nodes: root nodes located at the top of the tree and child nodes at each branch. Each node represents a condition, and the examples are divided into groups referring to the conditions (child nodes). Finally, a Gini impurity is adopted to measure the impurity of each node. The function of Gini impurity can be expressed as follow:

$$G_i = 1 - \sum_{k=1}^{n} {P_{i,k}}^2 \tag{3-9}$$

where $P_{i,k}$ is the portion of examples that belongs to category k among all training examples in the ith node.

The cost function for classification:

$$J(k, t_k) = \frac{m_{left}}{m} G_{left} + \frac{m_{right}}{m} G_{right} \tag{3-10}$$

where $G_{left/right}$ shows the Gini impurity of the left/right branch, and $m_{left/right}$ reparents the number of examples in the left/right branch.

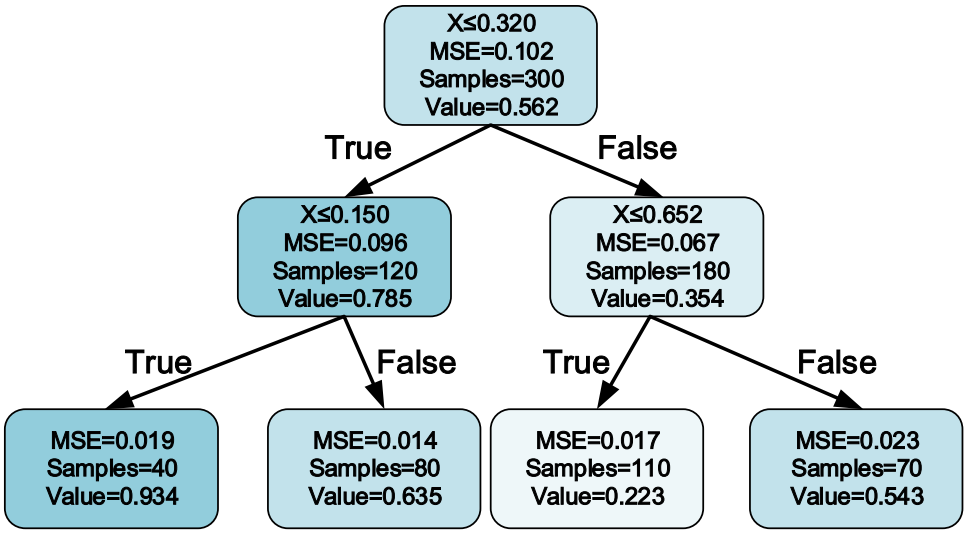


Figure 3-3. Example of the regression decision tree.

The cost function for regression:

$$J(k, t_k) = \frac{m_{left}}{m} MSE_{left} + \frac{m_{right}}{m} MSE_{right} \tag{3-11}$$

where

$$MSE_{node} = \sum_{i \in node} \left(\hat{y}_{node} - y^i\right) \tag{3-12}$$

$$\hat{y}_{node} = \frac{1}{m_{node}} \sum_{i \in node} y^i \tag{3-13}$$

When it comes to the random forest is a kind of ensemble learning method which consists of many decision trees, and the random forest is trained through bagging. It

makes predictions by taking a few trees' averages or mean output [255]. The accuracy of the prediction increases with the increase of the trees. Random forest solves the overfitting problem of the decision tree to achieve better prediction accuracy than the decision tree.

3.3.1.3 Gradient boosting regression tree

The Boosting algorithm is an ensemble learning algorithm that produces a stronger learner from a series of weak prediction models [256]. Among all boosting methods, Gradient Boosting Regression Tree (GBRT) is the most popular algorithm that is employed in multiple tasks. Normally, the GBRT algorithm contains three elements: a differentiable loss function for optimization, a squared error is adopted as the loss function for regression task; a weak prediction model to make a prediction, and the decision tree is used as a weak model in Gradient Boosting; and an additive model that can add all weak models together and minimize the losses [257], detailed steps of GBRT is presented in Algorithm 3-1.

Algorithm 3-1: Gradient Boosting Regression Tree (GBRT) algorithm

Input: Dataset $(x, y)_{i=1}^{N}$, where x the input features and y the target, $F(x)$ the prediction model, loss function $L(y, F(x)) = \frac{1}{2}(y - F(x))^2$, learning rate ν $(0 < \nu < 1)$, Iteration number M.

Output: $F_M(x)$.

1) Initialization. Set the $F_0(x) = \underset{\gamma}{argmin} \sum_{i=1}^{n} L(y_i, \gamma)$;

2) For m =1 to M:

a. Compute pseudo-residuals

$$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} = y_i - F(x_i)$$

for $i = 1,2 \cdots n$;

b. Fit a weak learner (regression tree in this case) to the r_{im} values (training model with data $\{(x_i, r_{im})\}_{i=1}^{n}$, and create terminal regions R_{jm} for $j = 1,2 \cdots J_m$;

c. Compute $\gamma_{jm} = \underset{\gamma}{argmin} \sum_{x_i \in R_{ij}} L(y_i, F_{m-1}(x) + \gamma)$

d. Update $F_m(x) = F_{m-1}(x) + \nu \sum_{j=1}^{J_m} \gamma_{jm}$.

Light Gradient Boosted Machine, or LightGBM for short, is a variant of a conventional gradient boosting algorithm [258]. Based on the gradient boosting decision tree algorithm, LightGBM has two improvements, gradient-based one-side sampling and exclusive feature bundling, to deal with large instances and features. These improvements dramatically speed up the training time and improve prediction accuracy in parallel.

3.3.2 Unsupervised learning

In contrast to supervised learning, unsupervised learning algorithms learn patterns from unlabelled data. Instead, the system must discover the patterns by itself [259]. Typical unsupervised learning algorithms include clustering, anomaly detection and dimensionality reduction.

3.3.3 Semi-supervised learning

Labelled training examples are usually difficult to obtain, and researchers often need to handle datasets with a large amount of unlabelled data. The semi-supervised learning algorithm is developed to deal with partially labelled data, and many semi-supervised algorithms employ a combination of supervised and unsupervised learning methods. The semi-supervised method will first classify the unlabelled data using unsupervised clustering methods, and then a supervised Machine Learning model is used to fine-tune the model.

Most of the tasks to be solved in this thesis are supervised learning, especially regression supervised learning, while unsupervised/semi-supervised learning algorithms are adopted to improve the model's performance.

3.3.4 Performance metrics for machine learning algorithms

Performance metrics evaluate the performance of machine learning models, and different performance metrics are employed depending on whether a classification task or a regression task. In machine learning and data science, ground truth is employed to represent the real value of output, while prediction means the predicted value generated by the machine learning models.

3.3.4.1 Performance metrics for regression models

To assess the performance of the regressor, the following four performance metrics are used in most cases: Mean Absolute Error (MAE), Normalized MAE (nMAE),

Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), Normalized RMSE (nRMSE), and R^2. As for metrics MAE, MAPE, RMSE, and nRMSE, a smaller value indicates a better prediction performance. The detailed formulas are as follows:

(1) MAE:

$$MAE = \frac{\sum_{i=1}^{N} |y_i - \hat{y}_i|}{N} \tag{3-14}$$

(2) nMAE:

$$nMAE = \frac{\sum_{i=1}^{N} |y_i - \hat{y}_i|}{N\bar{P}} \tag{3-15}$$

(3) MAPE:

$$MAPE = \frac{\sum_{i=1}^{N} |(y_i - \hat{y}_i)/y_i|}{N} \times 100\% \tag{3-16}$$

(4) RMSE:

$$RMSE = \sqrt{\frac{(\sum_{i=1}^{N} [y_i - \hat{y}_i]^2)}{N}} \tag{3-17}$$

(5) nRMSE:

$$nRMSE = \frac{RMSE}{\bar{y}} \tag{3-18}$$

(6) R^2:

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{3-19}$$

where y_i is the ground truth value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean value of the data points, SS_{RES} is the sum squared regression error and SS_{TOT} is the sum squared total error. The value of R^2 is between 0 and 1, while 1 means a perfect prediction as the predicted values are the same as the ground truth values.

3.3.4.2 Performance metrics for classification models

Evaluating a classifier is more complicated than a regressor; the common method is analysing the confusion matrix [243], as shown in Table 3-1. There are four combinations of the confusion matrix: TP represents True Positive, indicates the

number of positive examples classified accurately; FP means False Positive, indicates the number of actual negative examples classified as positive; FN represents False Negative, shows the number of actual positive examples classified as negative; TN means True Negative which shows the number of negative examples classified accurately [260]. Based on the confusion matrix, more concise metrics are developed to reveal more information about the prediction result: Accuracy, Recall, Precision, F-measure, and the receiver operating characteristic (ROC) curve.

Table 3-1. Confusion Matrix.

	Actual Positive	Actual Negative
Predicted Positive	True Positive (TP)	False Positive (FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

(1) **Accuracy.** Accuracy represents the ratio of correctly classified instances to the total number of instances; it tests the classifier's efficiency. The equation for accuracy is:

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \tag{3-20}$$

However, there are two drawbacks to Accuracy. Firstly, this metric may be misleading in the unbalanced dataset; secondly, the false predicted values are not used.

(2) **Recall**. Also named sensitivity, or the true positive rate, is the ratio of positive instances correctly detected by the classifier. The equation for Recall is:

$$Recall = \frac{TP}{TP+FN} \tag{3-21}$$

(3) **Precision.** Precision is the correctly predicted positive cases made by the classifier. The equation for Precision is:

$$Precision = \frac{TP}{TP+FP} \tag{3-22}$$

(4) **F-measure.** F-Measure is a metric which combines Recall and Precision and is widely used in prediction. Furthermore, it states the equilibrium between Precision and Recall:

$$F - measure = \frac{2 \times Recall \times Precision}{Recall+Precision} \tag{3-23}$$

The value of the F-measure is between 0 and 1, while a high F-measure score reparents a good classification performance.

3.3.5 Remarks

Based on the descriptions in the above sections about supervised learning, unsupervised learning, and semi-supervised learning, a comparison of these three machine learning algorithms is summarized in Table 3-2. From this table, the distinctive characteristic of the supervised learning method is that the training dataset should be well-labelled; hence the simple-structure model improves its performance by gaining experience from previous data. Since the model is well-trained with a large amount of data, the accuracy of supervised learning is normally the highest among the three methods. Unsupervised learning does not require labelling the training data and enables the model to find all kinds of unknown patterns in data by itself so that the unsupervised learning method can handle a large amount of data in real time. However, the result evaluated by the model cannot be verified due to a lack of transparency. The semi-supervised learning method only labels a small part of the data, and the small amount of training data can significantly improve inaccuracy. The choice of a specific machine learning method depends on the availability of labelled data and the exact problem to be solved.

Table 3-2. Comparison among supervised learning, unsupervised learning, and semi-supervised learning.

	Supervised Learning	**Unsupervised Learning**	**Semi-supervised Learning**
Type of Problem	Regression and Classification	Clustering and Association	Reward-based
Input Data	Labelled data	Unlabelled data	Partially labelled data
Process	Create a target function to map the input and output variables	Only input data is used for creating a model	Combine two process methods
Computational Complexity	Simpler	Computational complex	Depending on specific algorithms
Accuracy	Higher	Lesser	Lesser
Example Algorithms	Linear Regression, Decision Tree, K-Nearest Neighbours	K-means clustering, Principal Component Analysis	Self-Training, Semi-supervised Support Vector Machine

3.4 Deep Learning Foundation

Deep learning is an important subset of machine learning, and it employs a layered structure machine learning algorithm named Artificial Neural Network (ANN) [261]. Typical Deep Learning structures include Deep Neural Networks (DNN), Deep Belief Networks (DBN) [262], Graph Neural Networks (GNN) [263], et cetera. A typical ANN has an input layer, an output layer, and several hidden layers. Furthermore, the components inside an ANN contain neurons, weights, biases, transfer functions, and activation functions. A DNN represents a complex ANN with many hidden layers; the complex structure of the DNN enables it to learn the complex non-linear relationships between inputs and outputs. Depending on applications, DNNs can be further divided into Multi-Layer Perceptron (MLP) [264], CNN [265], and Recurrent Neural Network (RNN).

3.4.1 The basic structure of an artificial neural network

The biological neurons inspire the design of ANN, and a biological neuron is a particular cell in animal brains to transmits information by generating short electrical impulses. Although the single neuron behaves, high computation ability can be performed with the network combined by billions of biological neurons [266]. As the elementary units in an ANN, an artificial neuron is the mathematical model of the biological neuron [267]; an artificial neuron receives one or more binary inputs and processes a binary output (or activation). Like a biological neural network, an ANN is constructed by connecting many artificial neurons to transmit information from the input to the output. The basic structure of an artificial neuron is shown in Figure 3-4, and an artificial neuron contains inputs, weights, bias, and activation function. Inputs $x_1, x_2, \cdots, x_m$ are the real values a neuron receives from database or previous neurons. And each input x_i has a corresponding weight w_i, expressing the importance of the respective input to the output. Bias b is a constant term that adjusts the output along with the weighted sum of the inputs to the neuron. As shown in Equations (3-22) and (3-23), normally, an artificial neuron receives multiple inputs from other neurons, and

each input is multiplied by its associated weight, then the transfer function adds them up and passes the sum to an activation function f to obtain the output of the neuron.

$$y_{in} = x_1 \cdot w_1 + x_2 \cdot w_2 + x_3 \cdot w_3 \dots x_m \cdot w_m + b = \sum_i^m x_i \cdot w_i + b \quad (3\text{-}24)$$

$$y_{out} = f(y_{in}) \quad (3\text{-}25)$$

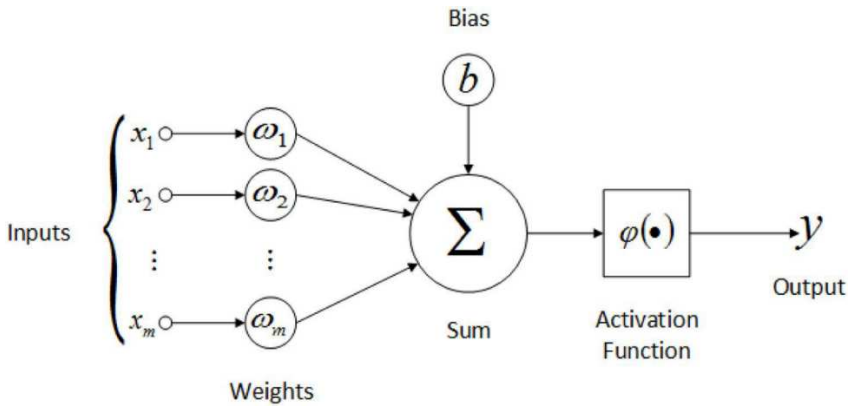


Figure 3-4. The structure of a simple artificial neural network (Adopted from [268]).

Although the structure of a single artificial neuron is simple, a complex ANN is obtained when combining millions of artificial neurons. Figure 3-5 shows the simplest ANN, and the ANN model contains one input layer, one output layer, and three hidden layers. The input layer is the leftmost layer of an ANN, and the neurons in the input layer are called input neurons. This layer receives initial data from the external database and brings the data to the subsequent layers to further processing. In contrast, the output layer is the rightmost layer that outputs the computation result of the ANN, and normally the output layer contains a single neuron for the regression tasks and several neurons for the classification tasks. These layers between the input layer and the output layer are the hidden layers, the inputs and outputs of these hidden layers are unknown to the researchers, and the purpose of the hidden layer is to perform nonlinear transformations of the inputs to the ANN. When an ANN contains a stack of hidden layers, this ANN is called a deep neural network (DNN). Training an ANN is divided into two steps: forward propagation and backpropagation [241].

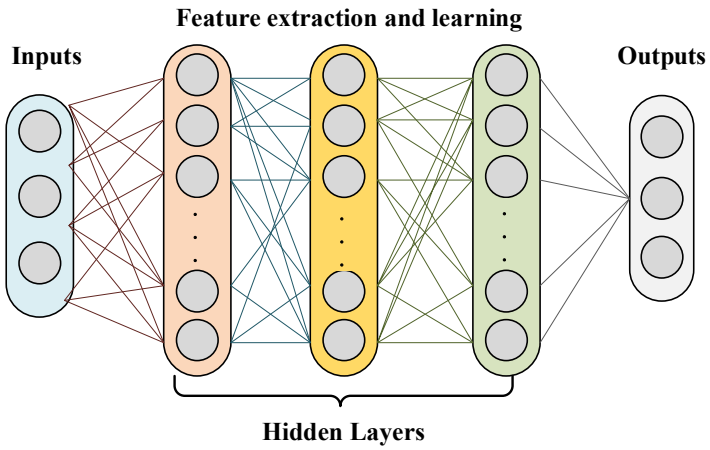


Figure 3-5. The structure of Multi-Layer Perceptron.

3.4.1.1.1 Forward propagation

The input layer obtains the input x and propagates the information through hidden layers, and produces output y. The output of neurons in layer l is:

$$a^{(l+1)} = f(W^{(l)}a^{(l)} + b^{(l)}) \tag{3-26}$$

where f is the activation function of the lth layer, and W and b are the weight matrix and bias of the lth layer. A loss function is adopted to measure the error between the ground truth (actual) output and the output generated by the model. Normally, a mean square error (MSE) is used as a loss function:

$$J(W, b; x, y) = \frac{1}{2}\left\|h_{W,b}(x) - y\right\|^2 \tag{3-27}$$

where $h_{W,b}(x)$ is the activation of the last layer.

3.4.1.1.2 Backpropagation

After the network's output error is computed, a backpropagation is used to update the model parameters generated in forwarding propagation. The backpropagation algorithm aims to compute the gradient descent of the network error computed in Equation (3-27) concerning each model parameter [241], and then the model

parameters (weights and biases) are tweaked to reduce the prediction error. The weights and bias of lth layer are updated by using a gradient descent method:

$$W^{(l)} = W^{(l)} - \alpha \frac{\partial J(W,b)}{\partial W^{(l)}} \tag{3-28}$$

$$b^{(l)} = b^{(l)} - \alpha \frac{\partial J(W,b)}{\partial b^{(l)}} \tag{3-29}$$

where α is the learning rate. The model parameters are updated by repeating the forward propagation and backpropagation process until the cost function minimises.

3.4.2 Hyperparameters

As the structure of a DNN is complex, there are many hyperparameters to tune. Typical hyperparameters include activation functions, regularization, loss function, and optimization. The following will introduce the most common hyperparameters and hyperparameter-tuning approaches.

3.4.2.1 Activation functions

In DNN, activation functions are always added to the network to provide nonlinear properties, and these activation functions can help the DNN learn complex mapping functions rather than simple linear regression. Activation functions adopted in this thesis are introduced in Figure 3-6.

1) **Linear function**. The linear function is the simplest activation function; the formula is:

$$f(x) = ax + c \tag{3-30}$$

The equation shows that the activation is proportional to the input, and the function's gradient is constant and equal to a. However, since the gradient is independent of the input, the updating factors of biases and weights during the backpropagation process will be equal. As a result, the error cannot be minimized during the training process, and the neural network model cannot extract features from the training data.

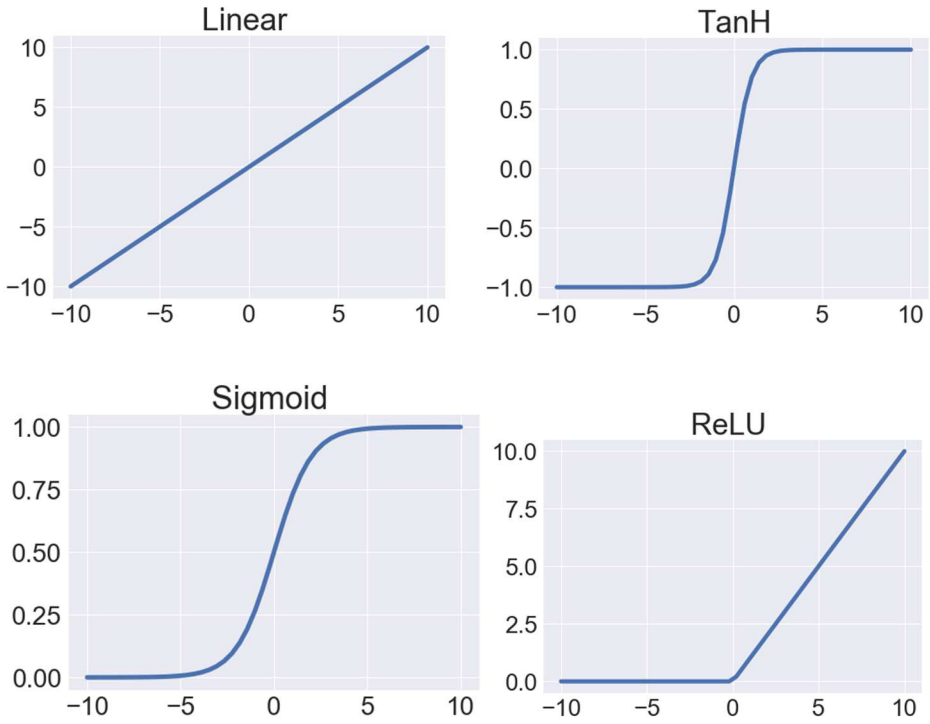


Figure 3-6. Linear, TanH, Sigmoid, ReLU activation functions.

2) **Rectified linear unit (ReLU):** ReLU function is a typical activation function. The mathematical equation of ReLU is $y = \max(0, x)$. ReLU is linear for positive values and 0 for all negative values.
3) **Sigmoid:** As a non-linear activation function, the output of the Sigmoid activation function is non-lear, and the value is between 0 and 1. Moreover, the Sigmoid curve is a smooth "S" shape curve and is differentiable continuously at any point. These characteristics help neural networks extract non-linear correlations between inputs and outputs efficiently. The equation for the Sigmoid activation function is:

$$f(x) = \frac{2}{1+e^{-x}} \tag{3-31}$$

4) **TanH.** TanH activation function is similar to the Sigmoid function as introduced above. Nevertheless, different from ReLU, TanH is a symmetric curve around the origin. The function is expressed as:

$$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1 \tag{3-32}$$

The value of the TanH function is between -1 and 1. When looking at the gradient, the gradient of TanH is much steeper than Sigmoid. This property prevents the gradients move in a certain direction and makes TanH well-adopted in applications rather than Sigmoid.

3.4.2.2 Regularization

Overfitting is one of the most serious problems when training deep neural networks: the model performs well with the training data but poorly with new data. The method to reduce the test error is called regularization.

3.4.2.2.1 ℓ_1 and ℓ_2 regularization

Both ℓ_1 and ℓ_2 Regularization added a parameter norm penalty $\Omega(\theta)$ to the objective function J to limit the capacity of neural network models [241]. These kinds of regularizations can be summarized as parameter norm penalties and can be expressed as:

$$\tilde{J}(\theta; x, y) = J(\theta; x, y) + \alpha\Omega(\theta) \tag{3-33}$$

where θ is the model parameter, α is the penalty weight. ℓ_2 Regularization is also called Ridge Regression or Tikhonov regularization [269]. $\Omega(\theta)$ of ℓ_2 Regularization is equal to $\Omega(\theta) = \frac{1}{2}\|w\|_2^2$. Hence, the regularized objective function $\tilde{J}$ that only considers w as parameter is defined as:

$$\tilde{J}(w; x, y) = J(w; x, y) + \frac{\alpha}{2}w^T w \tag{3-34}$$

The gradient of the regularized objective function $\tilde{J}$ is:

$$\nabla_w \tilde{J}(w; x, y) = \alpha w + \nabla_w J(w; x, y) \tag{3-35}$$

Furthermore, the weights can be updated as follows:

$$w \leftarrow (1 - \epsilon\alpha)w - \epsilon\nabla_w J(w; x, y) \tag{3-36}$$

It is observed that after $\boldsymbol{\ell_2}$ Regularization, a new weight decay term, shrinks the weights by a constant factor at each training step. Hence, the function of the $\boldsymbol{\ell_2}$ Regularization is to constrain a neural network's weights.

In term of $\boldsymbol{\ell_1}$ Regularization, the parameter norm penalty term $\Omega(\theta)$ is $\Omega(\theta) = \|w\|_1$. Hence the regularized objective function $\tilde{J}$ is:

$$\tilde{J}(w; x, y) = J(w; x, y) + \alpha\|w\|_1 \tag{3-37}$$

The gradient of the regularized objective function $\tilde{J}$ is:

$$\nabla_w \tilde{J}(w; x, y) = \alpha sign(w) + \nabla_w J(w; x, y) \tag{3-38}$$

where $sign(w)$ is the sign of w applied elementwise. The function of $\boldsymbol{\ell_1}$ Regularization is very different from $\boldsymbol{\ell_2}$ Regularization does not contribute to each weight but the gradient of the objective function instead. $\boldsymbol{\ell_1}$ Regularization uses a $sign(w)$ function to output binary weights from 0 to 1 to decrease the feature number in a large dimension dataset.

3.4.2.2.2 Dropout

Dropout is another regularization approach that is different from other regularization methods mentioned above. The core idea of the dropout is the dropout probability p: it means that each neuron inside the neural network (output neurons are excluded) has p probability of being ignored during each training step [270]. The outputs of these neurons dropped in this training step equal to 0.

3.4.2.3 Loss function

The loss function also called the objective function, measures the error between the predicted and actual values. For different tasks, different loss functions are selected. As for regression tasks, Mean Absolute Error (MAE), Mean Squared Error (MSE), Mean Squared Logarithmic Error (MSLE), and Huber Loss are usually used. For classification tasks, Binary Cross-Entropy Loss, Categorical Cross-Entropy Loss,

Hinge Loss, and Kullback-Leibler Divergence Loss (K-L Loss) are adopted as the loss functions.

3.4.2.4 Optimization

The optimizer determines how the network will be updated based on the loss function. It implements a specific variant of stochastic gradient descent (SGD). A comparison of the performance of different optimization algorithms is presented in Figure 3-7; Adam and RMSProp have the fastest convergence speed and good convergence quality, while Adagrad has a fast convergence speed but pool convergence quality, and SGD has the slowest convergency speed but the best convergence quality.

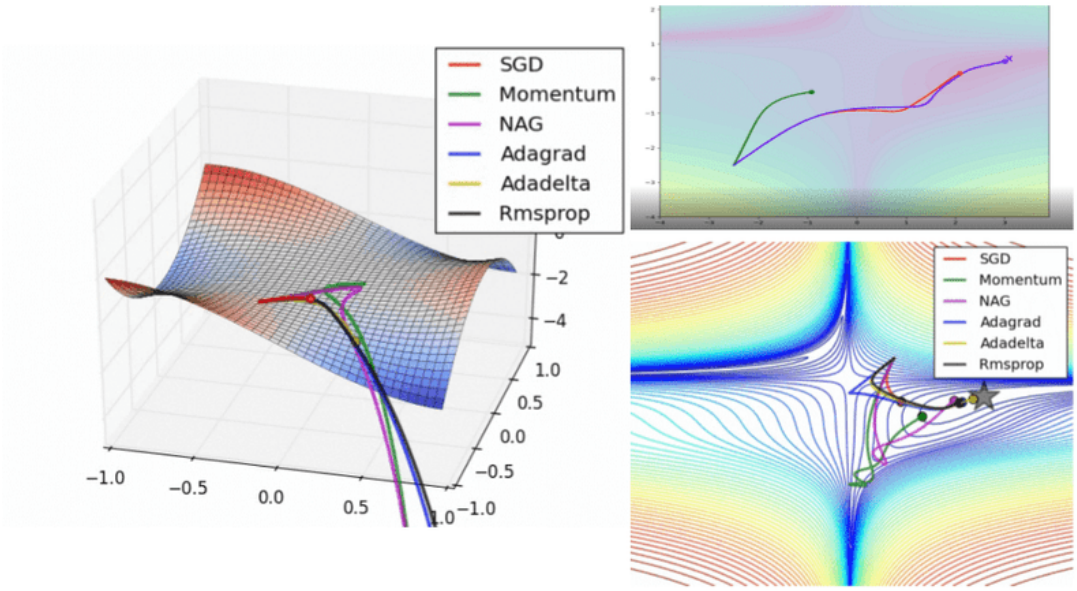


Figure 3-7. Comparison of the performance of different optimization algorithms [271].

3.4.2.4.1 Stochastic gradient descent

Recall Equations (3-28) and (3-29), which show the gradient descent process; it is concluded that gradient descent is an algorithm to find the optimal solution that minimizes the objective function. However, the main disadvantage of the traditional gradient descent algorithm is that it is inefficient and wastes a large amount of time as it should utilize all training datasets to calculate the gradient descent at each training

step. SGD only selects an instance of data from the training data and then calculates the gradient descent of the selected instance. The most significant advantage of SGD is that the computation time of each iteration will not increase with the increase of the training dataset size. Hence, the converging speed of SGD is considerably faster than the traditional gradient descent method.

3.4.2.4.2 Gradient descent with momentum

Momentum optimization obtains the inspiration of physics that a ball that rolls down a slope will pick up momentum quickly. However, traditional gradient descent and SGD algorithms will go down the slope with very small and fixed steps, taking a long time to reach the bottom. Furthermore, these optimizations do not consider previous gradients. A variable named v which represents the velocity, indicates the speed and direction of the parameters moving across the parameter space. In momentum optimization, it considers previous gradients. It will accumulate an exponentially decaying moving average of past gradients and continue moving in their direction [241]. Moreover, a parameter β which ranges from 0 to 1 represents exponential decay:

$$v \leftarrow \beta v - \epsilon \nabla_\theta J(\theta) \tag{3-39}$$

The above equation contains two parts, the first term is the gradient retained from previous training, and the second is the same as the standard SGD algorithm. The movement of the gradient can be decomposed into two components along w_1 and w_2 direction (see Figure 3-8), while w_2 is aligned with the ideal path, and w_1 is the vector that is orthogonal to w_1. When past gradients are accumulated, their components, along w_1 are cancelled out while their components, along w_2 are added up. So, in other words, the past gradients are used for acceleration. Moreover, the larger β related to ϵ, the more effect previous gradients have on the current direction. Finally, the parameters are updated with v:

$$\theta \leftarrow \theta + v \tag{3-40}$$

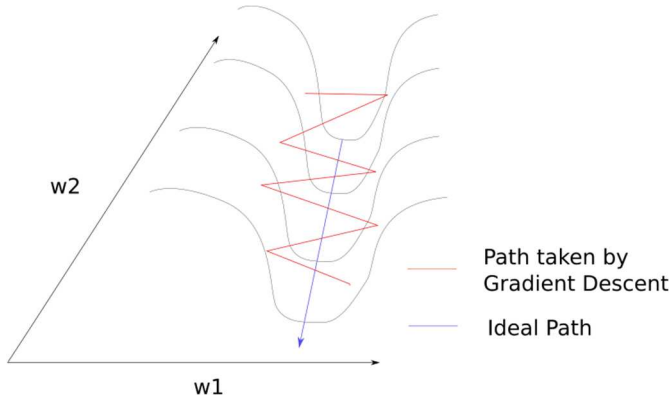


Figure 3-8. Comparison of the path taken by gradient descent and ideal path (Adopted from [272]).

3.4.2.4.3 RMSProp

RMSProp, which refers to Root Mean Square Propagation, is an optimization algorithm that can correct the gradient descent direction to reach the global optimum faster. The technology RMSProp utilizes, called adaptive learning rate, enables each parameter to obtain a different learning rate. RMSProp introduces an exponentially decaying average term that accumulates the most recent gradients:

$$r \leftarrow \rho r + (1-\rho)\nabla_\theta J(\theta)) \odot \nabla_\theta J(\theta) \tag{3-41}$$

The exponentially decaying average term will increase steadily as training steps increase. As a result, the original learning rate ϵ is decreasing over time:

$$\theta \leftarrow \theta - \frac{\epsilon}{\sqrt{r+\delta}} \odot \nabla_\theta J(\theta) \tag{3-42}$$

In this equation, δ is a smoothing term to avoid division by zero.

3.4.2.4.4 Adam optimization

Adam represents Adaptive Moment Estimation; this optimization algorithm combines momentum optimization and adaptive learning rates discussed above. Equations (3-43) and (3-44) are two decay terms similar to Momentum and RMSProp. (3-45) and (3-46) help boost s and r at the beginning of the training. Finally, the parameters are updated by Equation (3-47).

Update biased first-moment estimate:

$$s \leftarrow \rho_1 s + (1 - \rho_1)\nabla_\theta J(\theta) \tag{3-43}$$

Update biased second moment estimate:

$$r \leftarrow \rho_2 r + (1 - \rho_2)\nabla_\theta J(\theta) \odot \nabla_\theta J(\theta) \tag{3-44}$$

Correct bias in the first moment:

$$\hat{s} \leftarrow \frac{s}{1-\rho_1^t} \tag{3-45}$$

Correct bias in the second moment:

$$\hat{r} \leftarrow \frac{r}{1-\rho_2^t} \tag{3-46}$$

Compute update:

$$\theta \leftarrow \theta - \epsilon \frac{\hat{s}}{\sqrt{\hat{r}}+\delta} \tag{3-47}$$

3.4.3 Important neural network models

3.4.3.1 Convolutional neural network

CNN is a kind of Deep Learning model that replaces the traditional fully connected layer with convolutional layers in at least one of all layers [241]; it is normally applied in image recognition and image classification. The biological neurons inspire the development of CNN in the visual cortex [273]. These neurons have receptive fields that enable them to react to specific patterns [243]. These small patterns then are reconstructed in the brain to present more complex patterns to recognize large images. As shown in Figure 3-9, the simplest CNN model contains an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer.

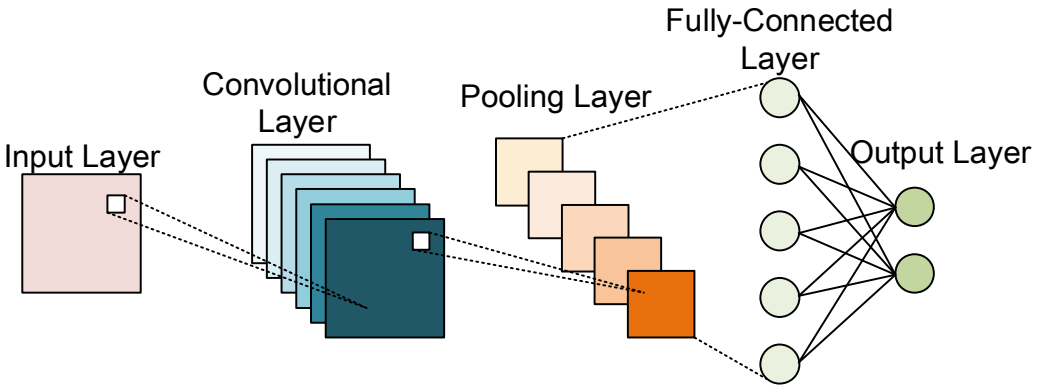


Figure 3-9. The structure of the convolutional neural network (Adopted from [265]).

To better understand the concept of CNN, it is compulsory to understand the convolution operation. Given the original function x, the convolution operation aims to obtain a new function which is the weighted average or smoothed estimation of x. And a weighting function $w(a)$ is sliding over time and multiplies with the original function:

$$s(t) = \int x(a)w(t-a)da \tag{3-48}$$

3.4.3.1.1 Convolutional layer

A convolutional layer is the core block of CNN models, and it is the first hidden layer that links with the input layer to extract different features from the input data.

Normally, the input data of the CNN model is 4D tensors of shape (samples, input height, input width, input channels) or (samples, input channels, input height, input width). In contrast to the MLP, a convolutional layer processes data only for its receptive fields [274]. The first convolutional layer can learn small local patterns such as edges, colours, and gradient orientation, and the next convolutional layer can assemble these low-level features into larger-level features such as eyes, noses, and ears. After the data passes each convolutional layer, the original data is abstracted to a feature map with the shape of (samples, feature map height, feature map width, and several filters). It is noticed that the index of the channel in a feature map stands for the number of filters [265]. Record the convolution operation function in (3-48); in a CNN, function x is referred to as inputs, the weighting function w is referred to as the

filter (kernel), and s is referred as the feature map. A 2D CNN example is shown in Figure 3-10; in this example, the filter size is (2×2); during the operation, the filter is sliding over the inputs, and the dot product is taken between the filter and the part of the inputs that are corresponding to the filter. Finally, a feature map will show the inputs' detailed features, including corners, edges, et cetera. Other parameters in a convolutional layer include Padding, Stride. Padding is adding layers of zeros to the inputs to avoid the pixels in the corner that cannot be used (border effect problem). Moreover, the step of the movement of the filter is called stride, and the stride can change the shape of the feature map.

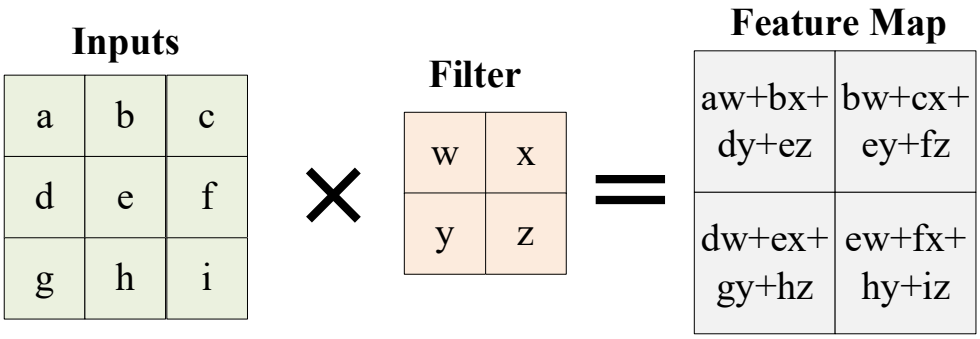


Figure 3-10. An example of 2-D convolution.

3.4.3.1.2 Pooling layer

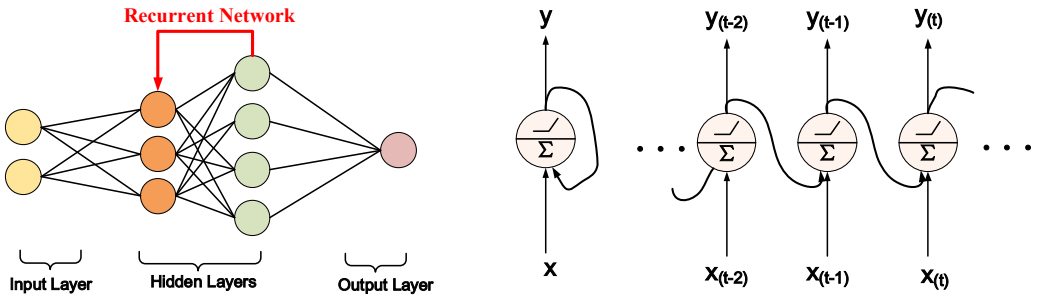
A pooling layer is always stacked after the convolutional layer. The pooling layer's purpose is to reduce the dimensions of the feature maps to reduce the computational load and the memory usage [243]. Similar to the convolutional layer, neurons in a pooling layer connect a part of neurons located in the receptive field in the previous layer. Moreover, the padding type and the stride should also be defined like the convolutional layer. Normally, two typical pooling layers are adopted: the max-pooling layer and the average pooling layer [275].

3.4.3.2 Recurrent neural network

A Recurrent Neural Network (RNN) is a special DNN that contains loops to enable the network to predict future events by using previous experiences [276]; see Figure

3-11 (a). Similar to the structure of traditional feedforward DNN, the only difference between RNN and DNN is that RNN has a connection that points backward [243] (Figure 3-11 (b)). By unrolling the recurrent neuron, it is found that at any time step t, the neuron receives inputs at t $x_{(t)}$ and outputs at previous time step $t-1$, $y_{(t-1)}$ (Figure 3-11 (c)). Hence, each recurrent neuron has two sets of weights:W_x for $x_{(t)}$, and W_y for $y_{(t-1)}$, and the output of the recurrent neuron is:

$$y_{(t)} = \phi(W_x^T x_{(t)} + W_y^T y_{(t-1)} + b) \tag{3-49}$$



(a) (b) (c)

Figure 3-11. (a) The structure of a recurrent neural network; (b) A recurrent neuron; (c) unrolled recurrent neurons through time (Adopted from [277]).

However, the traditional recurrent neural network has two serve issues: gradient vanishing and exploding problems and limited short-term memory problems [243]. The flow of the backpropagation starts from the output layer to the input layer, and the error gradient is propagated along the direction. However, the gradients will turn smaller and smaller as the backpropagation process moves to lower layers, and the weights of these lower layers remain unchanged. As a result, the iteration will never converge to an optimal point. This problem is the so-called gradient vanishing problem.

In contrast, the gradient sometimes grows bigger and bigger, which results in great weight updating, and the neural network diverges; this is exploding problem. Another problem is that conventional RNN contains a very limited short-term memory. This issue is caused due to long-term information being easily lost when travelling through all cells before arriving at the present cell. To solve these two problems mentioned

above, two novel RNN structures, Long Short-Term Memory (LSTM) [278] and Gated Recurrent Units (GRU) [279] networks, are introduced in the literature.

3.4.3.2.1 Long short-term memory and gated recurrent unit neural network

A recurrent Neural Network (RNN) is a special DNN that contains a hidden state to enable the network to predict future events using previous experiences. However, gradient vanishing and exploding problems [280] and limited short-term memory problems [165] limit the development of RNN for a long time. Unlike conventional RNN designed from short-term memory and the naive RNN has a poor performance for long sequences (Vanishing Gradient), LSTM and GRU can retain long-term and short-term information without much loss by introducing a memory cell. Moreover, LSTM and GRU have gates to help the memory cell regulate past information.

In GRU, the cell state is equal to the output at time step t, GRU has only two gates, a reset gate $\boldsymbol{r_t}$, and an update gate $\boldsymbol{z_t}$ [279] (shown in Figure 3-12 (b)). The reset gate is responsible for determining the combination of current inputs $\boldsymbol{x_t}$ with previous cell state $\boldsymbol{h_{t-1}}$, it has a sigmoid function to regulate the output value between 0 and 1, and the gate can identify how relevant the information from previous steps $\boldsymbol{h_{t-1}}$. Update gate $\boldsymbol{z_t}$, which is like forget gate in LSTM, is developed to decide whether update cell state or not. The expressions for the above two gates are:

$$\boldsymbol{z_t} = \sigma(\boldsymbol{W_z} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}]) \tag{3-50}$$

$$\boldsymbol{r_t} = \sigma(\boldsymbol{W_r} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}]) \tag{3-51}$$

where σ stands for sigmoidal activation; $\odot$ represents element-wise multiplication. $\boldsymbol{W_z}, \boldsymbol{W_r}$ are the weight matrices. With the information from $\boldsymbol{r_t}$, $\boldsymbol{h_{t-1}}$ and $\boldsymbol{x_t}$, the candidate value from the current state $\widetilde{\boldsymbol{h}}_{\boldsymbol{t}}$ is calculated. Finally, the current cell state is determined by $\boldsymbol{h_{t-1}}$ and $\widetilde{\boldsymbol{h}}_{\boldsymbol{t}}$ to remind the previous state or update to a new value:

$$\widetilde{\boldsymbol{h}}_{\boldsymbol{t}} = \phi(\boldsymbol{W} \cdot [\boldsymbol{r}_{\boldsymbol{t}} \odot \boldsymbol{h}_{\boldsymbol{t-1}}, \boldsymbol{x}_{\boldsymbol{t}}] \tag{3-52}$$

$$\boldsymbol{h}_{\boldsymbol{t}} = (1 - \boldsymbol{z}_{\boldsymbol{t}}) \odot \boldsymbol{h}_{\boldsymbol{t-1}} + \boldsymbol{z}_{\boldsymbol{t}} \odot \widetilde{\boldsymbol{h}}_{\boldsymbol{t}} \tag{3-53}$$

where ϕ represents tanh activation.

The LSTM model was firstly proposed in 1997 [27]. As shown in Figure 3-12 (a), in LSTM, the hidden state in traditional RNN is replaced by the memory cell $\boldsymbol{C}_{\boldsymbol{t}} \in \Re^{h \times 1}$ (h denotes the number of hidden units) and three gates, i.e., the input gate $\boldsymbol{I}_t \in (0,1)^{h \times 1}$, the forget gate $\boldsymbol{F}_{\boldsymbol{t}} \in (0,1)^{h \times 1}$, and the output gate $\boldsymbol{O}_{\boldsymbol{t}} \in (0,1)^{h \times 1}$. The output of the previous time step $\boldsymbol{h}_{\boldsymbol{t-1}} \in (-1,1)^{h \times 1}$ and the input sequence of the current time step $\boldsymbol{X}_{\boldsymbol{t}}$ are adopted as the input of the gates. The sigmoid activation function σ controls these gates $(\cdot)$: the information is reserved when the activation output is close to 1, and the information is eliminated when the activation output approaches 0. As for the memory cell $\boldsymbol{C}_{\boldsymbol{t}}$, a candidate memory cell $\widetilde{\boldsymbol{C}}_{\boldsymbol{t}} \in (-1,1)^{h \times 1}$ is computed at first. The only difference between $\widetilde{\boldsymbol{C}}_{\boldsymbol{t}}$ and the gates are that $\widetilde{\boldsymbol{C}}_{\boldsymbol{t}}$ utilizes a Tanh activation function $tanh(\cdot)$ ranging from -1 to 1. Finally, the memory cell $\boldsymbol{C}_{\boldsymbol{t}}$ is generated by combining $\widetilde{\boldsymbol{C}}_{\boldsymbol{t}}$ and $\boldsymbol{I}_{\boldsymbol{t}}$ the previous memory cell $\boldsymbol{C}_{\boldsymbol{t-1}}$ with $\boldsymbol{I}_{\boldsymbol{t}}$ and $\boldsymbol{F}_{\boldsymbol{t}}$, where $\boldsymbol{I}_{\boldsymbol{t}}$ decides how much data from $\widetilde{\boldsymbol{C}}_{\boldsymbol{t}}$ is useful, and $\boldsymbol{F}_{\boldsymbol{t}}$ decides how much information from the old memory cell is retained. The detailed formulas are presented as follows:

$$\boldsymbol{I}_{\boldsymbol{t}} = \sigma(\boldsymbol{W}_{\boldsymbol{xi}}\boldsymbol{X}_{\boldsymbol{t}} + \boldsymbol{W}_{\boldsymbol{hi}}\boldsymbol{h}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\boldsymbol{i}}) \tag{3-54}$$

$$\boldsymbol{F}_{\boldsymbol{t}} = \sigma\big(\boldsymbol{W}_{\boldsymbol{xf}}\boldsymbol{X}_{\boldsymbol{t}} + \boldsymbol{W}_{\boldsymbol{hf}}\boldsymbol{h}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\boldsymbol{f}}\big) \tag{3-55}$$

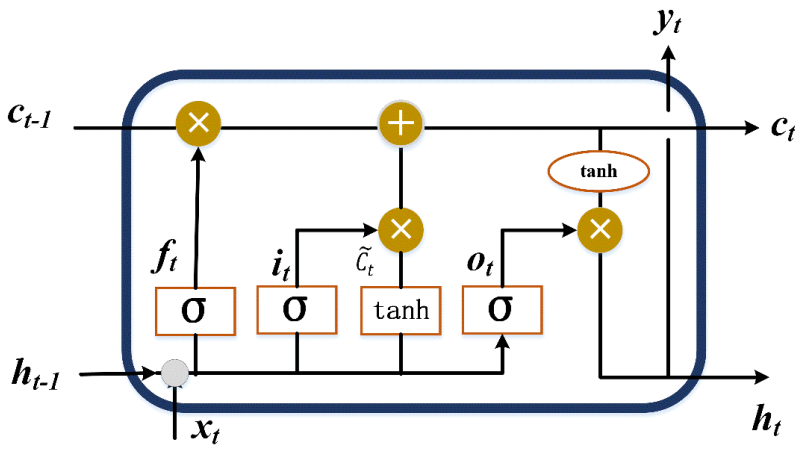
$$\boldsymbol{O}_{\boldsymbol{t}} = \sigma(\boldsymbol{W}_{\boldsymbol{xo}}\boldsymbol{X}_{\boldsymbol{t}} + \boldsymbol{W}_{\boldsymbol{ho}}\boldsymbol{h}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\boldsymbol{o}}) \tag{3-56}$$

$$\widetilde{\boldsymbol{C}}_{\boldsymbol{t}} = tanh(\boldsymbol{W}_{\boldsymbol{xc}}\boldsymbol{X}_{\boldsymbol{t}} + \boldsymbol{W}_{\boldsymbol{hc}}\boldsymbol{h}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\boldsymbol{c}}) \tag{3-57}$$

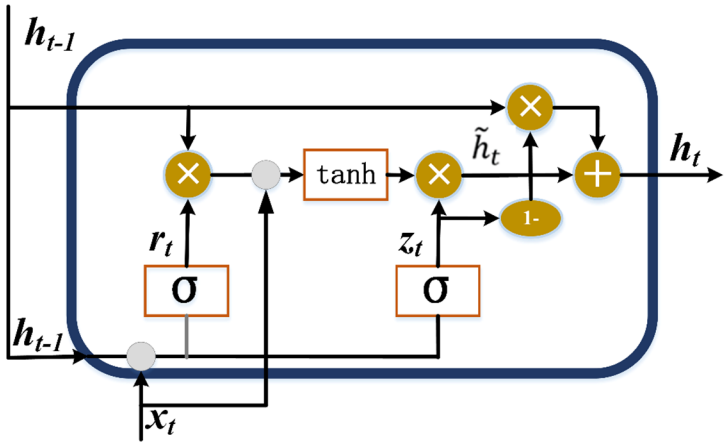
$$\boldsymbol{C}_{\boldsymbol{t}} = \boldsymbol{F}_{\boldsymbol{t}} \odot \boldsymbol{C}_{\boldsymbol{t-1}} + \boldsymbol{I}_{\boldsymbol{t}} \odot \widetilde{\boldsymbol{C}}_{\boldsymbol{t}} \tag{3-58}$$

$$\boldsymbol{h}_{\boldsymbol{t}} = \boldsymbol{O}_{\boldsymbol{t}} \odot tanh\,(\boldsymbol{C}_{\boldsymbol{t}}) \tag{3-59}$$

where $\boldsymbol{X_t} \in \Re^{\eta \times 1}$,$\odot$ represents element-wise multiplication; $\boldsymbol{W_{xi}}, \boldsymbol{W_{xf}}, \boldsymbol{W_{xo}}, \boldsymbol{W_{xc}} \in \Re^{h \times \eta}$ (η is the number of examples) ; $\boldsymbol{W_{hi}}, \boldsymbol{W_{hf}}, \boldsymbol{W_{ho}}, \boldsymbol{W_{hc}} \in \Re^{h \times h}$ are the weight matrices; and $\boldsymbol{b_i}, \boldsymbol{b_f}, \boldsymbol{b_o}, \boldsymbol{b_c} \in \Re^{h \times 1}$ are the bias parameters.



(a) LSTM-RNN



(b) GRU-RNN.

Figure 3-12. The structure of (a) LSTM-RNN; (b) GRU-RNN.

3.4.3.2.2 Bidirectional long-short term memory

The main disadvantage of the conventional LSTM model is that it can only utilize the information from the past. A BLSTM was proposed in 1997 [28] to overcome this drawback. As shown in Figure 3-13, unlike the unidirectional LSTM, BLSTM can utilize both previous and future information with two separate LSTM layers, i.e., a forward LSTM layer that passes information from the past to the future a backward

LSTM layer that passes information from the future to past. As the data collected by the smart meter is sequence data in the time domain, the BLSTM model is especially suitable for processing such data for the following reasons. Firstly, the amount of input a BLSTM model can reach is larger than the standard LSTM model, and the rich information gives BLSTM a much higher data representation capability [29]. Secondly, the BLSTM models do not follow the recursive procedure, and this characteristic enables these models to make predictions on stochastic and intermittent data with high accuracy.

In a BLSTM structure, given a minibatch input $\boldsymbol{X'_t} \in \mathcal{R}^{\eta \times d}$ (d is the sequence size of each example), the forward and backward hidden states at time step t, i.e., $\overrightarrow{\boldsymbol{H}_t} \in \Re^{\eta \times h}$ and $\overleftarrow{\boldsymbol{H}}_t \in \Re^{\eta \times h}$ can be expressed as:

$$\overrightarrow{\boldsymbol{H}_t} = \phi(\boldsymbol{X'}_t \boldsymbol{W}_{xh}^{(f)} + \overrightarrow{\boldsymbol{H}}_{t-1} \boldsymbol{W}_{hh}^{(f)} + \boldsymbol{b}_h^{(f)}) \tag{3-60}$$

$$\overleftarrow{\boldsymbol{H}}_t = \phi(\boldsymbol{X'}_t \boldsymbol{W}_{xh}^{(b)} + \overleftarrow{\boldsymbol{H}}_{t+1} \boldsymbol{W}_{hh}^{(b)} + \boldsymbol{b}_h^{(b)}) \tag{3-61}$$

where $\boldsymbol{W}_{xh}^{(f)}, \boldsymbol{W}_{xh}^{(b)} \in \Re^{d \times h}$, $\boldsymbol{W}_{hh}^{(f)}, \boldsymbol{W}_{hh}^{(b)} \in \Re^{h \times h}$ represent the weights of the model; and $\boldsymbol{b}_h^{(f)}, \boldsymbol{b}_h^{(b)} \in \Re^{\eta \times h}$ are the biases of the model. Then, by integrating the forward and backward hidden states, the hidden state is obtained as $\boldsymbol{H_t} \in \Re^{\eta \times 2h}$. Finally, H_t is fed to the output layer to compute the output of the BLSTM block $\boldsymbol{O_t} \in \Re^{\eta \times q}$ (q is the number of outputs):

$$\boldsymbol{H_t} = \left[\overrightarrow{\boldsymbol{H_t}}^T \quad \overleftarrow{\boldsymbol{H_t}}^T\right]^T \tag{3-62}$$

$$\boldsymbol{O_t} = \boldsymbol{H_t} \boldsymbol{W_{hq}} + \boldsymbol{b_q} \tag{3-63}$$

where $\boldsymbol{W_{hq}} \in \Re^{2h \times q}$ is the weight; and $\boldsymbol{b_q} \in \Re^{\eta \times q}$ is the bias of the output layer.

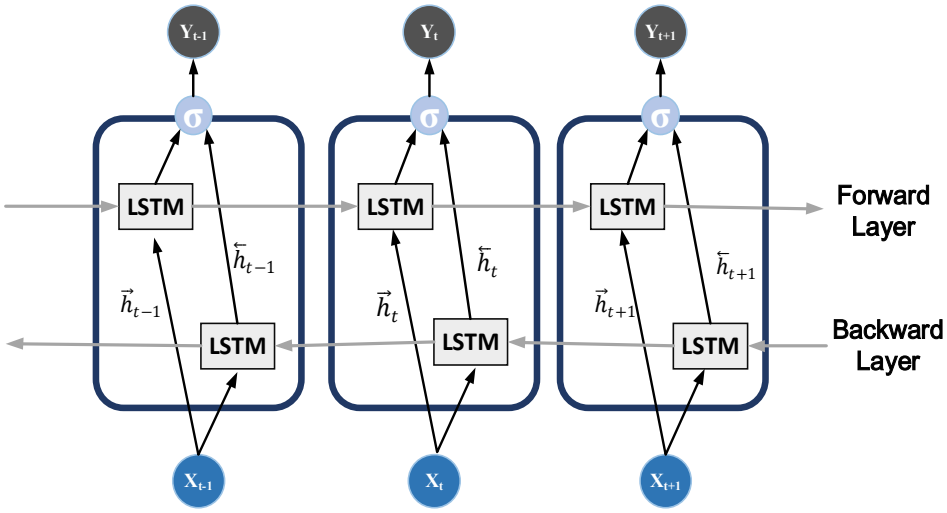


Figure 3-13. The structure of Bidirectional LSTM.

3.5 Chapter Summary

This chapter presents a comprehensive overview of data analytics and data mining methods for smart meter data. Foundation knowledge includes machine learning, deep learning, information theory, and data pre-processing are introduced in detail. Machine learning is divided into supervised machine learning and unsupervised machine learning algorithms, depending on training. Meanwhile, deep learning includes MLP, CNN, RNN, and RNN can be further divided into LSTM, GRU, and BLSTM, respectively. Machine learning and deep learning are important tools for the decision-making models to be constructed in the following chapters, while information theory quantifies the privacy loss of the smart meter data. In summary, the rich information provided in this chapter paves the way for building complex models in the next chapters.

Part II Privacy

Chapter 4 A Privacy-Preserving Multi-Channel Smart Metering System

4.1 Introduction

4.1.1 Motivation

AMI, or smart metering system, is the backbone of the modern smart grid. AMI is an integrated system that combines electricity equipment, a communication network, renewable generation, and the energy management centre to enable two-way information flow between the consumers and the utility [281]. Apart from basic billing and energy consumption information, the AMI also enables remote consumption control, TOU pricing, load forecasting, energy theft detection, etc. [282].

In AMI, smart meters are edge sensors inside the consumers' houses, monitoring the energy consumed inside the consumer's house. However, the large volume of high granular power consumption data collected by the smart meter brings new challenges to consumers' privacy. Several recent research works [86, 235, 283] and government reports [284, 285] have highlighted that the existing smart meter roll-out plans are against legal frameworks regarding privacy and data protection, such as GDPR [108]. One reason is that the highly granular data contains sensitive personal information, and the residents' behaviours and activities can be inferred from the smart meter data by implementing data mining algorithms. As a result, the rollout plans of several countries, such as Germany and the Netherlands, have been deferred due to privacy concerns [286, 287]. Hence, new technical solutions are desired to alleviate the privacy concern and better conduct the smart meter roll-out plan.

As a utility-centric system, the current smart metering system only has a single channel to transmit the energy consumption data, which at times causes the

consumers' personal information, which is disclosed to the energy utility in the process, to be larger than required. Referring to the literature presented in Chapter 2, relevant technical solutions include rechargeable battery [288], renewable energy storage system [57], noise-adding [289], homomorphic encryption [79], data anonymization [85], data aggregation [80], and data downsampling [290]. These approaches are discussed based on the following dimensions:

- **Integrity**. The correctness and accuracy of the metadata must be guaranteed [45].
- **Computational complexity**. Considering the large scale of the smart metering system, the technical solution requires high computational complexity and would cause serious computation overhead.
- **Privacy-by-design**. The privacy should be an integrated part of the smart metering system without any external device/hardware [110]. Moreover, a technical solution with an external device would increase the costs of the energy supplier or the consumers.
- **Latency**. The technical solution should have no/little latency for real-time grid operation and management purposes.
- **Environmentally friendly**. The technical solution should not harm the environment or increase CO2 emissions.

Rechargeable battery [288] employs a rechargeable battery to mask the original load curve by the charging/ discharging process; the power consumption curve detected by the adversary is a flattened curve, so the personal activities are hidden. However, this solution is not a privacy-by-design solution, and the consumers must purchase the rechargeable battery/ energy storage system (which costs thousands of pounds [71]) only for privacy purposes; who should undertake the cost is a tough question to answer. Moreover, rechargeable battery has an environmental downside. It would pollute the environment, which is against one of the original purposes of rolling out the smart meter: meeting the carbon-neutral target from the Paris Agreement [291] and the EU Energy Efficiency Directives [292].

Noise-adding [289] obfuscates the original metadata by adding noise, e.g., Gaussian Noise, Laplace Noise, so the data inferred by the adversary has been distorted. Although the noise cancels out if enough readings are added together or after a long period, this method also introduces latency into the control loop. Grid operation and management functions such as demand response and state estimation, which require no latency, would be influenced. Furthermore, the attacker/adversary may denoise the distorted data once he/she knows the specific noise-adding method employed by the smart meter.

Data anonymization [80] aims to hide privacy by replacing the real smart meter identification with pseudonyms. However, several researchers [45, 54] have pointed out that attackers can still infer a consumer's location by linking other databases, such as the record of blackout events.

Data aggregation [80] builds some aggregators to aggregate the neighbouring smart meter records and then sends the overall data to the energy utility. The typical aggregation approach introduces a Trust Third Party (TTP) to operate these data aggregators, while TTP could be a potential honest-but-curious adversary who could also cause information disclosure [45]. Moreover, involving TTP in the smart metering system could increase the cost and reduce the system's reliability. To better regulate the data aggregator, new laws and regulations are desired to enforce the operation of the data aggregators.

Data down sampling [290] reduces the sensitivity of the personal data by downsampling the interval resolution of the power consumption data. Referring to [104], data resolution is the main factor determining information disclosure. Most appliances can be detected from smart meter data at one minute, while only occupancy information can be inferred with half-hourly data. Although low-resolution data reduces the sensitivity, this method disables functions that require high-frequency measurement, e.g., Time-of-Use Tariff, grid operation and management, and demand response.

Homomorphic encryption [79] is a cryptographic technique that enables linear operations, e.g., addition and manipulation of the encrypted smart meter data; homomorphic encryption is combined with other technical solutions such as data aggregation to ensure confidentiality of the data during the transition. However, homomorphic encryption is computationally intensive, requiring large storage space and a server with high computation ability. The existing homomorphic encryption-based solution is only simulated on a small-scale smart metering system in the virtual environment, and it still has a long way to go for a large-scale application in practice.

By analysing the existing methods, the knowledge gaps and the limitations of the existing privacy-preserving solutions are summarized as follows:

1) Existing methods only transmit a single interval resolution data, while the data granularity required by different stakeholders varies a lot.
2) Some solutions require installing extra hardware device or high computational algorithm/encryption, which increases the cost and have a downside to the environment. Hence, the stakeholders involved are not sufficiently engaged in the smart metering system as they cannot see long-term benefits.
3) Whilst many solutions provide a strong privacy guarantee, the basic functionalities such as TOU tariff, grid management, and value-added services are sacrificed.

4.1.2 Objective of the chapter

The objective of this chapter can be summarised as follows:

1) Define the threat/model which could invade the consumer’s privacy.
2) Investigate the critical functions provided by the smart metering system and define the minimum required data to achieve the corresponding functions.
3) Develop a privacy-preserving smart metering system that satisfies both privacy requirements and functionalities.
4) Quantify the aggregation size's privacy boundary and the metadata's interval resolution.

4.1.3 Scope of the research

To better develop a privacy-preserving smart metering system from a power system discipline scope, the scope of this research is identified as follows:

1) The smart meter in this thesis is assumed to be the certified device that would record the energy consumption honestly following the protocols. Since if an adversary can control the smart meter, the adversary would bypass any data protection technologies. Moreover, the smart meter is assumed to be temper evident to guarantee the correctness of the reading. In addition, the smart meter is also assumed to have the capacity to store and secure the long-term keys for the smart meter and protect its privacy [293]. In practice, there could be malicious clients such as the energy theft, which would try to temper the meter reading to steal electricity from the energy utilities. Such malicious clients are out of the scope of this thesis and will be investigated in future work.
2) This research only focuses on power system parameters recorded by the smart meter, e.g., active power, reactive power, cumulative energy consumption, voltage, bills, and TOU tariff, while other personal information such as service contract, bank account, phone number, the email account is out of the scope.
3) The entities involved in this research are limited to residential consumers, energy suppliers, distribution network operators, and third-party service providers; other entities such as industrial consumers, commercial consumers, and government are out of the scope.
4) Detailed cyber-attack scenarios on the smart grid and the smart meter are out of the scope of this research.

4.1.4 Contribution

In this chapter, a privacy-preserving smart metering system is developed based on the approaches from [76, 85, 88] to the combined use of existing data aggregation and data down-sampling techniques to design a privacy-preserving smart metering system. The system follows an operational and ethically driven trade-off strategy and model,

which could increase the functionalities of current smart metering devices in smart grids whilst ensuring that digital privacy intrusion is minimised and protected if not appropriately governed. In addition, the system provides three different communication channels for data collection to enable diverse data granularity transmission to other stakeholders, with each channel also providing required functionalities (time-of-use billing, grid operation and management, and value-added services). The contributions of this chapter include:

1) The trust and the adversary/attack models are developed to determine the potential privacy risks in the existing smart metering system.
2) A privacy-preserving smart metering system which enables three communication channels to transmit different granular data is designed.
3) A data mining algorithm utilized by the adversary is designed to evaluate the privacy boundary of the smart meter data.

4.1.5 Structure of the chapter

The chapter is organized as follows: In Section 4.1, the main contributions of this chapter are proposed: a trade-off strategy is discussed with a proposed new smart metering system model to support it. Section 4.2 develops the trust model and the threat/adversary model. The privacy functionality trade-off strategy is demonstrated in Section 4.3. Furthermore, in Section 4.4, a multi-channel smart metering system is designed, referring to the trade-off strategy. In Section 4.4, the privacy boundary is detected via the data mining algorithms. The privacy risk of the proposed system is evaluated in Section 4.6. The chapter summary is drawn in the last section of the chapter.

4.2 Threat/Adversary Model

This section starts with identifying the notion of privacy employed in this thesis, then based on the privacy definition, categories of the threat/adversary, the purpose and

target of the threat model, as well as the data mining algorithms that the adversary may use is introduced.

4.2.1 Notion of privacy

Unlike the Internet or traditional communication network, the smart grid is a cyber-physical-social system that involves a heterogeneous power system structure, multiple stakeholders, and many endpoints [45]. Hence, the notions of privacy must be introduced at the beginning better to analyse the privacy risks in the existing system. There are three significant distinctive definitions of privacy in turns of smart meter and smart metering system: ethical privacy, statistical privacy, and cryptographic privacy.

4.2.1.1 Ethical privacy

Ethical privacy is a privacy definition from an ethical aspect, which is a foundation of human rights in society. One of the famous formulations proposed by S. Warren and L. Brandeis in 1890 describes privacy as 'the right to be let alone [294]'. This definition proposed that everyone has his/her area of activity without constraint, coercion, and even surveillance. An individual has freedom of choice to decide the data to be shared and the freedom to decline any unauthorised access to smart meter data, the individual also needs to know the purpose of data collection, and he/she can reject inappropriate use of data. More specific to the field of the smart grid, four dimensions of privacy is summarized by the National Institute of Standards and Technology (NIST) from U. S. Department of Commerce [284], which include the privacy of personal information, privacy of the person, privacy of personal behaviour, the privacy of personal communications, detailed definition of each dimension is listed as below:

1) **Privacy of personal information**: This definition is the broadest researched dimension, and it is under the protection of the GDPR. Personal information indicates all information related to an individual that can reveal the consumer's

physiological, physical, economic, cultural, or social identity. Privacy of personal information gives the consumer the right to control where, how, when, to what, and to whom to share his/her personal information, and the consumer should also be given authority to access, modify, and correct the information that has been shared, the safety of the information should also be ensured [295].

2) **Privacy of the person**: Privacy of the person is the right to maintain the integrity of an individual, which includes physical requirements, health problems, and required medical devices.
3) **Privacy of personal behaviour**: This dimension indicates the right to keep the individual's behaviour from being shared with others; the personal behaviour contains the individual's activities and choices.
4) **Privacy of personal communications**: This point highlights that the individual should have the right to communicate with others without being monitored [295].

As a utility-centric system, the existing smart metering system does not provide consumers with enough personal autonomy and freedom. In the current smart metering system, the consumers passively share their data without options to decide what granularity of data to be transmitted and whether they would like to share the data with the energy utility. Moreover, current data protection laws/regulations such as GDPR mainly cover the first dimension-privacy of personal information, while other points are also important to guarantee privacy. Hence, all four dimensions of ethical privacy should be considered to construct a user-centric system.

4.2.1.2 Statistical privacy

Statistical privacy indicates that a dataset will not reveal an individual's private information. Differential Privacy (DP) is the widest adopted statistical privacy notion proposed by C. Dwork in 2006 [296]. Given a database that contains the information from many participants, the main idea of DP is that if the effect of a single substitution is small enough, then the output of an enquiry from the database will not contain an individual's personal information. Referring to Smart Metering Implementation Programme: Review of the Data Access and Privacy Framework published by DBEIS

in 2018 [12], the electricity consumption data which is modified via aggregation or anonymisation securely will no longer be personal data since the modified data has no connection to a single domestic property. Referring to the definition of statistical privacy, private data and public data can be defined as below:

1) **Private information**: the individual smart meter data or the data can still be identified as a single domestic property after data processing.
2) **Public information**: the energy consumption data processed by temporal/spatial aggregation, anonymisation or other methods to disconnect the link with individual property. After data processing, data mining algorithms cannot detect an individual's personal information.

4.2.1.3 Cryptographic privacy

Cryptographic technology does not solve the privacy problem directly, but it is an extremely helpful tool to guarantee the privacy of information and confidentiality during data transition and storage in an open environment [297]. Cryptographic privacy means no information leakage – the information that is revealed by an algorithm is limited to the information that can be inferred from this algorithm. The adversary learns nothing about which consumer is communicating with each other even after many rounds of communication [45]. In contrast, the privacy-preserving system guaranteed by differential privacy allows quantifiable metadata leakage after many communication rounds [298].

4.2.2 Trust model

Before identifying the threat/adversary model of the smart metering system, the trust model needs to be defined first. Normally in a privacy-aware system, the trusted parties in the system do not require further privacy protection. In contrast, untrusted parties could either be malicious (perform any algorithm for stealing, corrupting, and modifying data [299]) or honest but curious (they will follow the communication protocol honestly, but they would keep all information received from other parties and

try to infer individual measurements [300]). The smart metering system builds trusting relationships between the energy consumer and stakeholders (ES, DNO). A trusted energy consumer means that the energy consumer can provide correct energy consumption and bills data without tampering with the reading and record of the smart meter, while a trusted entity means the entity can access the consumer's energy consumption data authorized by the consumer for legitimate purposes. In the modern smart metering system, the smart meter is considered fully trusted since a Trusted Platform Module (TPM) is embedded to verify the correctness of reading and bills; the energy consumer and stakeholders, including ES and DNO, are also trusted entities, while TP is an honest-but-curious stakeholder.

Although ES and DNO are trusted entities, the amount and granularity of the collected sensitive data are strictly regulated by data regulation laws such as GDPR (2018) [301], Data Protection Act in the U.K. (2018) [302] and previous EU Data Protection Directives (1995) [303]. In GDPR, the data minimisation principle requires that organizations cannot collect more information than they need, and the originations are also required to identify the minimum amount of personal information (interval, period) that can fulfil their purpose [301]. Data Protection Act also highlights that the organizations who want to access personal data must obey strict regulations named 'data protection principles' [302]. 'Data protection principles' require the organizations to use the personal data for specified, explicit purposes, and the collected data is limited to only what is necessary [302]. More specific to the smart metering system, the data access and privacy framework published by BEIS requires DNO to provide a detailed report to state the format, the purpose, the period, and the target consumers of the smart meter data before DNO is approved to access smart meter data [12]. Moreover, OFGEM [41] requires DNO to aggregate the smart meter data to remove the individual identity from the dataset. Hence, even though ES and DNO are trusted entities, collecting the whole fine-granular data without a sufficient and careful justification is also a privacy invasion that infringes these regulations/principles [304].

4.2.3 Threat/adversary model

In this subsection, the threat/adversary models $\mathcal{A}$ which have the potential to invade consumers' privacy are identified. The following introduces the adversary's purposes and internal and external adversaries who want to infer personal information.

4.2.3.1 Purpose/target of the adversary

Figure 4-1 shows a household electricity consumption profile with an interval resolution of 15 min, which is the regular sampling frequency of the smart meter. From the figure, high-resolution energy consumption data collected by the smart meter can reveal detailed electricity activities by implementing data mining algorithms such as NILM. Based on the smart meter data and NILM algorithm, the following information is obtained by the adversary:

1) **Appliance usage information**: The operation status of the household appliances, such as air conditioner, dishwasher, kettle, wash machine, and refrigerator.
2) **Presence/absence**: Indicate whether the resident is present at the house or away for a holiday. When the resident is away, most electronic appliances are turned off, and few activities are detected (it should be noticed that the refrigerator will keep turn-on/turn-off automatically; hence non-refrigerator events are used to determine presence/absence).
3) **Event/ behaviours**: Events in the house, such as breakfast, lunch, dinner, party, shower, and playing video games.
4) **Sleep cycle**: Detect the time when the resident goes to sleep and s/he wakes up.

Although NILM algorithms have many beneficial applications, such as increasing the energy awareness of the consumers and helping the operator implement demand response, the NILM algorithms can also be utilized by the potential adversaries to reveal individual lifecycles, which increase potential surveillance possibilities posing physical, financial, and reputational risks [284]. The potential purposes for the adversaries to collect the smart meter data can be summarized into four categories:

commercial purpose, illegal purpose, legal purpose, and family members' usage (See Table 4-1).

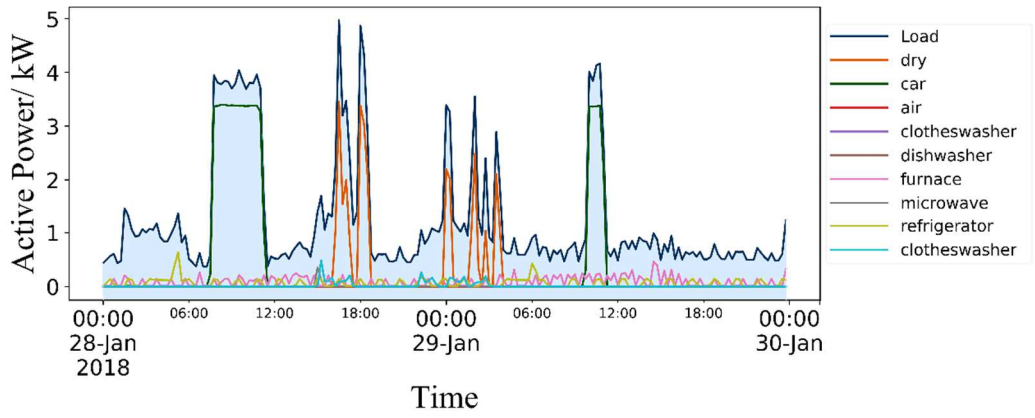


Figure 4-1. Example of household load profile, with detailed appliance usages (Data source: Pecan Street Dataport [117]).

4.2.3.1.1 Commercial purpose

Private/commercial companies are the main beneficiaries of the smart meter data, and these companies have strong motivation to extract appliance usage information for directed advertisements [305]. It is even possible for commercial companies to identify the specific brand of the electronic appliance used by the consumer by implementing the NILM algorithm. Based on this detailed appliance information inside the consumer's house, these commercial companies may send customers targeted advertisements for the electronic appliances that need to be replaced/repaired/upgraded [104]. In addition, the insurance companies can adjust the credit rating for the consumers who have bad electricity usage habits, e.g., the consumer who always levels the heater/stove on when he is away from home; as a result, this consumer has a higher possibility to suffer from fire hazard than those people who will always turn off all appliance before leaving home. Although the commercial companies can profit from the high granularity of smart meter data, the consumer may have strong disapproval and resistance to such unauthorized actions since their in-home activities are exposed, and their privacy is invaded [305].

4.2.3.1.2 Illegal purpose

The smart meter data inferred by the adversaries will also be abused for illegal purposes, as stated in Table 4-1. Burglar/thief can find the occasion that the house is empty by analysing the energy consumption curve. When only a refrigerator event is detected in the house for a period, the burglar/thief can confirm that no resident is inside the property [305]. As a result, the burglar/thief can determine their targets and break into the empty houses. Moreover, the stalkers may tap into an intermediate AMI node to monitor their victims' activities and behaviours inside the house.

4.2.3.1.3 Legal purpose

The government and legal organizations such as police officers also want to access the smart meter record for many purposes. One typical case is the police can detect drug production, illegal bitcoin production, and energy theft, the energy consumption for such activities is much higher than the normal consumers, e.g., producing drugs requires indoor growing systems which are equipped with fans and high lighting intensities lights, sometimes operate 24 hours a day [306]; the mining of the bitcoin also requires a large amount of electricity to run high-power computers [307] continually. In 2007, the Austin Police Department in the U.S. was authorized to access power consumption records without a search warrant [305]. Furthermore, the smart meter data can be adopted as evidence to verify the defendant's claims, e.g., whether the defendant stayed in the home at that time as he/she claimed.

4.2.3.1.4 Used by family members/ co-inhabitants

The family member/ landlord also has the potential motivation to monitor other members inside the property by analysing the electricity activities inside the room. For instance, the parents may be curious about what their children are doing inside their room, and the children could be punished once there are found playing video games or watching TV [308]. In addition, the landlord can use the smart meter data to double-check whether a certain electronic appliance is used properly. Such

surveillances seriously invade an individual's privacy or personal behaviour, as the individual has the right to be let alone, and both the children and the tenant try to keep their activities confidential [294].

Table 4-1. Summary of the purpose of the adversaries in the smart metering system.

Type of Usage	Purpose
Commercial purpose [104]	1. Advertising to the target consumers, e.g. Promoting electronic appliances by identifying the broken appliances that need to be renewed/repaired/updated inside the consumer's house. 2. Insurance level adjustment, e.g., reducing the insurance level of the consumers who always leave the heater/stove on when they are away from home.
Illegal purpose [305]	1. Burglar/theft to detect whether a house is occupied. 2. Stalkers monitor the lifecycle/behaviour patterns of their victims.
Legal purpose [306, 309]	1. The police detect illegal activities inside the property, e.g., drug production. 2. Verifying the defendant's claims, e.g., that he/she was 'at home all day'.
Family members/ co-inhabitants [305]	1. Family members monitor the activities of other members inside their room, e.g., parents check whether their children are studying or playing computer games. 2. Ensure the children have not locked inside the home alone. 3. The landlord monitors his/her tenant to investigate whether an appliance is overused (e.g., dishwasher).

4.2.3.2 Internal adversary

The adversary model is shown in Figure 4-2, which contains both internal adversaries and external adversaries. Internal adversaries/attackers indicate the threat/adversary inside the smart metering system. Whilst TPs in the smart metering system, which reparents non-licence third-party service provider/commercial companies, are considered the honest-but-curious adversaries, honest-but-curious/semi-honest adversary is widely used in smart grid/ smart meter privacy problems in the literature [76, 300, 310]. The definition of honest-but-curious/semi-honest adversary is shown below:

Definition 1 (Honest-But-Curious Adversary) [300]. *The honest-but-curious adversary represents a legitimate protocol member who will not deviate from the defined protocol but will attempt to study as much information as possible from received messages*.

The honest-but-curious adversaries will follow the communication protocol honestly without malicious actions and cannot obtain more information than they receive (honest). However, they would keep all information received from other parties and try to infer individual measurements (curious). In other words, all parties work properly to maintain the system's operation while maximising the chance of acquiring individual' privacy.

4.2.3.3 External adversary

The smart grid and the smart metering system highly rely on the wireless communication network, while the wireless communication channel is vulnerable to cyber-attacks from the external adversary as the channel is naturally a broadcast transmission medium [311]. External adversary indicates the adversary, which is not any stakeholder/sector inside the smart metering system.

A typical external adversary of the smart meter is an eavesdropper. The communication techniques used in the smart metering system, such as ZigBee, WiMax, WiFi and PLC, are vulnerable to cyberattacks that could lead to eavesdropping [312]. The malicious eavesdropper may eavesdrop on the communication channel between the smart meter and the energy utilities to obtain the energy consumption data recorded by the smart meter. Referring to [311], five potential communication channels can eavesdrop:

1) The consequence of the smart meter and the cellular tower is that some meter reading data is disclosed.
2) The consequence of the smart meter and the third parties is that some meter reading data is disclosed.
3) Between the cellular tower and the utility, a considerable amount of meter reading data is disclosed.
4) The consequence of the utility and the third parties is that most meter reading data is disclosed.

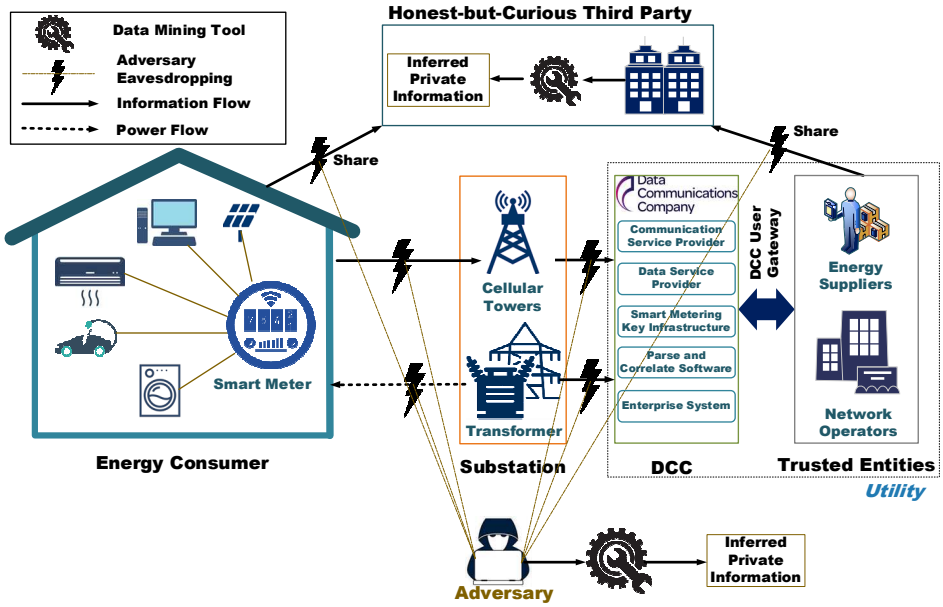


Figure 4-2. Adversary/attacker model in the smart metering system.

It should be highlighted that the resolution of the data eavesdropped on the communication is a standard/policy rate (15 min for the current system). Moreover, the overhead feeder between the houses and the distribution transformer is exposed to the public, and the adversary also can install sensors on the feeder to eavesdrop on the houses under the feeder.

Whilst the information obtained by the eavesdropper can either be plaintext or ciphertext. When the data is not encrypted before transmitting, the information obtained by the eavesdropper is plaintext which is easy to obtain the smart meter data without extra traffic analysis. When the smart meter data is encrypted or pre-processed before transmitting, the malicious eavesdropper can still find valuable information by monitoring a large amount of data since some words/characters remain the same after being encrypted [311]. Based on the discussion above, all adversaries are summarized in Table 4-2.

Table 4-2. Summary of threat/adversary.

Type of Adversary	Threat Scenario	Result	Threat Likelihood	Consequence
Internal Adversary	Semi-honest TP infer personal information	The majority of meter data shared with TP is disclosed	Likely	Catastrophic
External Adversary	Eavesdrop communication between SM and cellular station Eavesdrop communication between SM and TP	Some meter reading data is disclosed	Likely	Moderate
	Eavesdrop communication between cellular station and utility	A considerable amount of meter reading data is disclosed	Likely	Serious
	Eavesdrop communication between TP and utility	Majority of meter reading data is disclosed	Likely	Catastrophic
	Plug its meter into the feeder between SM and transformer	The aggregated meter reading under the feeder is disclosed	Likely	Minor

4.3 Privacy-Functionality Trade-Off Strategy

This section develops a privacy-functionality trade-off strategy based on the threat model introduced above, the existing private data regulation policy in the UK and EU and the compulsory functionalities.

4.3.1 Requirement For the Proposed Smart Metering System

As the privacy notions introduced above, the proposed smart metering system should satisfy both ethical privacy, statistical privacy, and cryptographic privacy. First, from an ethical aspect, the proposed system should be shifted from a utility-centric to a user-centric system, giving the consumer personal autonomy and freedom from undue surveillance. The consumer has a flexible choice in deciding whether they would like to share the data with other parties. The energy utilities and third parties are granted limited trust to achieve this target, and the energy consumers are given full control over their personal information. Detailed requirements for the proposed smart metering system are listed below:

1) Obtain the correct meter reading and bills from the smart meter.
2) The smart metering system to be designed should better fit the GDPR, which only collects the minimum data for required functions.

3) The system should be a privacy-by-design, not rely on external devices and hardware.
4) Prevent internal adversary access to individual data.
5) Prevent the smart meter reading be eavesdropped on by the external adversary.
6) Enables critical functions required by the stakeholders, especially TOU tariff, value-added service, and grid operation and management.

4.3.2 Compulsory functions

As required by GDPR, The European Commission identifies the 13 main functions of a smart meter and classifies them into five categories [313]. The most significant functions listed in [45] are billing correctness, grid operation and management, and additional consumer services. In addition, an emerging function is that of the TOU tariff. The interval resolution and categories of data for these critical functions are listed in Table 4-4 below.

4.3.2.1 Billing and Time-of-Use tariff

The smart meter's most vital function is providing accurate consumer billing. Any data protection method which influences the accuracy and the correctness of the billing is useless. The current sample interval of the smart meter is 15 minutes, but consumers do not need such high-frequency billings; monthly billing is enough [45]. The TOU tariff determines the electricity price during different periods. Consumers benefit from the TOU tariff by shifting their electricity usage habits to enjoy a cheaper bill, while the energy suppliers can also reduce the power plant capacity as a result [13].

Moreover, the TOU tariff can also increase the demand-side flexibility and contribute to increasing the penetration of renewable energy, so TOU is becoming the mainstream method for billing in the UK. With the installation of the smart meter, the TOU moves closer to the real-time pricing tariff, allowing it to represent the actual conditions [314] better. Introducing the TOU tariff increases the electricity price in peak periods and lowers it in off-peak periods.

4.3.2.2 Grid operation and management

In the past, the distribution network was not well monitored due to the limitation of communication and the metering infrastructure. Recently, distribution-level grid monitoring and management have attracted more attention due to the deep penetration of distributed renewable energy and the uncertainty caused by the electrical loads. The smart meter contributes to the smart grid by improving the efficiency and stability of the whole power system. The real-time two-way communication networks provided by the smart metering system can measure, analyse, and control the energy consumption data and further support the smart grid in implementing demand response services and power system estimation. For grid operators, the measurement of every individual household smart meter is not compulsory. Instead, they have more interest in regional aggregated data, such as measurement at the feeder or distribution level [73]. Such regional aggregated data can be used for feeder/distribution-level applications such as load forecasting, distributed renewable generation detection, and energy components analysis.

- **Load forecasting**: day-ahead load forecasting increases the predictivity of the distribution network. Electricity data with an interval between 15 minutes to 1 hour is required to make precise forecasting [315].
- **Renewable generation detection**: Existing distribution network is highly penetrated with renewable generation such as rooftop solar panels. Such renewable generation is highly fluctuating and difficult to predict generally, which introduces uncertainty in monitoring the distribution network. Hence, detecting these renewable energies will decrease the uncertainty of the network. In the literature [212, 316, 317], electricity data with intervals of 5-15 minutes is required to estimate solar energy generation.
- **Energy components analysis**: Analysing the load components under the feeder/distribution network can increase the visibility of the network and help the DNO better understand the real-time condition of the power system. From related

work [204, 318], feeder-level data with 1-30 minutes is used to train the machine learning models.

4.3.2.3 Value-added services

Consumers can order additional services provided by third-party service providers. The additional consumer services could be awareness (e.g., sending a warning for exceeding power) or scheduling and control (scheduling for controllable appliances, peak shaving) [319]. Demand-side response and NILM have received the most attention. Demand-side response [45, 320, 321] optimizes the strength of the grid and enhances the power quality by utilizing power plants, distributed generators, loads and energy storage. In demand response, consumers can also participate in the response process by accepting the bids provided by grid operators. Turning off appliances such as air conditioners and heaters would shed the load during peak time. NILM is a technique to disaggregate consumers' power consumption curve into individual appliance usage. The consumer can understand how electricity is consumed and better manage their home appliances to save energy and reduce carbon dioxide emissions [47]. Typically, value-added services require consumers to submit their energy data to a server; the server would use a pre-trained model to evaluate the data and send the results back to consumers. The difficulty exists in how to share personal data with TP while guaranteeing privacy at the same time. In [45], two privacy-preserving value-added schemes are proposed. The naive scheme down-samples the original data into multiple interval resolution data, referring to the requirement of different services. Then the different resolution data are sent to different TPs with a key [88]. The second solution enables services on consumers' devices (personal computers, mobile phones) via a HAN. However, TPs have the risk of revealing their models/algorithms.

4.3.2.4 Summary of data required by each function

Based on the discussion above, the data required to achieve billing, TOU tariff, grid management & operation, and value-added services are introduced in Table 4-3. For billing purposes, the frequent transmission of the power consumption data would put

consumers under the monitoring of the utility. For grid operation and management, although the utility requires high interval resolution data, it is unnecessary to access every individual's power consumption; aggregated data of an area (feeder or distribution network) is desirable. Most additional services provided by TP only require a specific part of the power consumption data (a certain period, a specific appliance power consumption, etc.), and these services are optional depending on the consumers' choices. Hence, in the proposed strategy, all TPs must obey the data minimization principle (explained below) and only collect the minimum data required with consumers' consent to complete the service.

Table 4-3. Summary of data granularity of different functionalities.

Functionalities	**Sampling Frequency Required**	**Data Required**
Billing	Low (weekly or monthly cumulative energy consumption)	Usage of every single household smart meter
Grid Operation and Management	High (between 5 min -1 hour for load forecasting and profiling)	Active/Reactive power, Voltage, Current, etc.
Value-Added Services	Depending on specific services	Depending on specific services
TOU tariff	High (15 min–1 hour)	The TOU price from the electricity market and the energy consumption duration of this period

4.3.3 Operation strategy

Given the scale of smart meter roll-out processes in countries and worldwide, the above risks and operational strategies could be dismissed or subordinated to utilitarian market logic, with the responsibility for their implementation and subsequent privacy protection of consumers (i.e., households) delegated to third parties, many of whom might not have privacy protection as a priority in their agendas. Moreover, and as stated before, there is a lack of clarity about such responsibilities. Furthermore, whilst smart grids could be conceived as necessary technologies to regulate the conduct of individuals in the societies [10], what could be more concerning is that privacy intrusion could also generate negative social consequences [11]. Consumers can be left powerless or socially isolated to devise strategies to counteract intrusion into their privacy, becoming mere means rather than ends [5].

It might be possible, however, for stakeholders to exert their creativity even in the face of privacy intrusion and existing regulations (i.e., GDPR directive) [5, 8, 322]. The creativity would help households comply with digital technologies established for them [322] whilst socially protecting or enhancing their sense of authentic household 'hood' [6]. To meet this, a trade-off strategy is proposed that attends to both the operational and ethical concerns for smart meters and smart grids raised in this chapter, and it utilizes a hybrid and soft strategy which combines the privacy design strategy (as introduced in 2.6) and privacy-functionality trade-off strategy. The hybrid strategy can be summarized as follow:

(1) Adopt data-oriented strategies (Minimise, Separate, Abstract, and Hide). The designed system only collects minimal personal data for specific functionalities (Minimise). In addition, the proposed system should enable a distributed framework; hence, consumers can utilize or store their data on their personal devices (Separate). Moreover, rather than sharing or transmitting high-resolution data with companies, only an abstract version of the data is shared to avoid revealing details of personal information (Abstract). Furthermore, end-to-end encryption technology should be utilized to guarantee privacy/confidentiality (Hide) better.
(2) Adopt process-oriented strategies (Inform and Control). Consumers should be informed what kind of smart meter data is collected and how their data is processed on a real-time base (Inform). Most importantly, the consumers should also have the right to choose the personal data to be collected, and they also have the right to select wanted functionalities (Control).
(3) Seek an optimal balance between privacy and functionalities. Privacy must not come at the expense of functionality; the proposed system should ensure all compulsory functionalities and seeks an optimal balance between privacy and functionalities.
(4) Maximize the retention of original facilities. The economic benefit of the smart meter is the common interest of both energy suppliers and consumers, and the

privacy-preserving scheme should not be costly (due to the installation of additional devices or sensors such as an energy storage system).

4.4 Multi-Channel Smart Metering System

4.4.1 The preliminaries

The hardware complexity of the smart meter in the proposed system remains the same as the existing smart meter. The smart meter has basic storage and computation ability to save power consumption and calculate the bills. Assume the area involves a smart meter group $\boldsymbol{SM} = \{SM_1, SM_2, \cdots SM_i, \cdots SM_N\}(i \in [1, N]$. The smart meter can measure power consumption with interval T (normally 15 minutes), marked as P_{SM_i}. The smart meter data are encrypted to prevent consumers from modifying the power consumption data. There is no backdoor when the smart meter is manufactured, so manufacturers or energy suppliers cannot illegally access the smart meter data, and the DCC monitors all data transmission between consumers and the utility.

4.4.2 Overall system

Based on the trade-off strategy illustrated in Section 4.3, the proposed multi-channel smart metering system is shown in Figure 4-3. The system components are consumers, DCC, ES, DNO, TP, and the Aggregator. Moreover, in contrast to conventional smart metering systems that can only transmit a single temporal resolution trace, this novel scheme contains three communication channels supporting multi-temporal resolution data. These three channels are:

- A high-frequency aggregated data channel transmits high-frequency aggregated data measured at the distribution level substation.
- A TOU billing channel to send dynamic TOU price information to smart meters and bills to the ES monthly.
- A value-added service channel to transmit selected data to support value-added services provided by TP.

The smart meter in the scheme plays the role of the assistant processor rather than the information sender and receiver; it has the basic computation ability to calculate billing inside the house rather than sending individual power consumption near real-time. The detail of each channel is illustrated as follows.

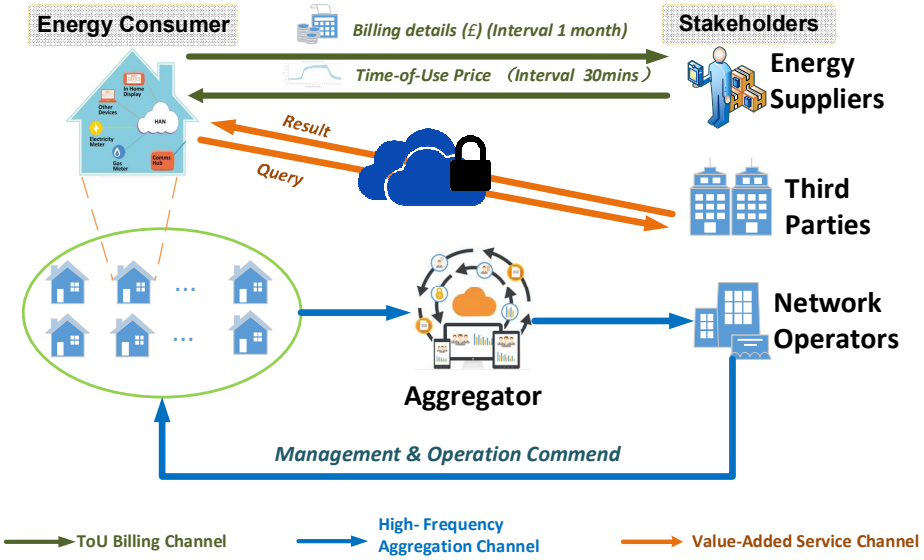


Figure 4-3. Multi-channel smart metering system.

4.4.3 High-frequency aggregation channel

Referring to the data access regulation published by OFGEM [41] and BEIS [12], the smart meter data used by the DNOs should be pre-processed (such as data aggregation or data anonymization) to remove the individual features from the dataset; the processed data which disconnects the correlations with individual identify will no longer be regarded as private information. In addition, the maximum sampling frequency of the smart meter is limited to 10s, referring to the BEIS specification [25]. However, the increasing penetration rate of renewable energy generation, electric vehicles, and energy storage systems requires a meter reading at a much higher rate in the future to manage better and control the distribution network [54]. To resolve the contradiction between privacy and DNO's requirement, a high-frequency aggregation channel is designed to enable grid operation and management functionality required by the DNO without collecting individual data directly. Instead of transmitting individual energy consumption data, this channel transmits the aggregated power

consumption data to the DCC. The aggregated smart meter data enables the DNO better monitor the condition of the LV distribution network; significant improvements in LV grid management and operation are supposed to be achieved given such high granular data, which include:

- Increase the visibility of the load components and renewable energy generations.
- Fast response to the faults ensures energy supply to the consumers.
- Optimize the design and planning to accommodate new connections better.

Denote α as the total number of smart meters under the data aggregator. At each timestep t, the active power of the data aggregator $f_{\mathcal{P}}^{agg}(t)$ are calculated:

$$f_{\mathcal{P}}^{agg}(t) = \sum_{i=1}^{\alpha} P_{SM_i}(\mathrm{t}); t = 1,2,\cdots,T \tag{4-1}$$

A comparison of power consumption of a single house and aggregated power consumption is presented in Figure 4-4. As shown in the figure, with the increasing aggregation level, the power consumption curve becomes smoother, and the details of individual appliance signatures become difficult to extract. Considering the physical structure of the distribution network and the two-way communication, two aggregation schemes are proposed in this research: the physical aggregation approach and the informatic aggregation approach. To better illustrate the proposed schemes, two concepts need to be introduced at first, which are information flow and power flow:

- **Information flow**: The information flow shows how the information is exchanged between entities through communication channels.
- **Power flow**: The power flow indicates the flow of electric power in an interconnected power system. The transmission line or feeder links different energy entities such as generators, transformers, and energy consumers.

As a Cyber-Physical system, the smart grid contains information and the power flow. The physical aggregation scheme aims to capture the aggregated data by installing meter devices on electricity equipment such as the feeder and the transformer, while

the informatic aggregation scheme aims to aggregate the smart meter data via adding the digital signal transmitted from the smart meters.

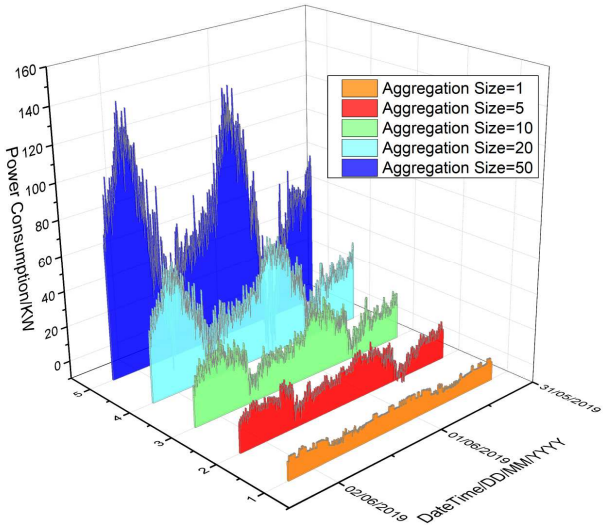


Figure 4-4. Single house power consumption versus different aggregation sizes of power consumption.

4.4.3.1 Physical aggregation approach

The main idea of the physical aggregation approach is to employ smart meters/sensors installed at different distribution network locations to obtain the multi-level aggregated measurements (LV distribution level and feeder level). DNOs have installed Supervisory Control and Data Acquisition (SCADA) and Phasor Measurement Unit (PMU) to monitor the operating conditions of the High Voltage (HV) and the Medium Voltage (MV)[1] network. However, such SCADA systems are not yet implemented in LV distribution networks [323]. Although recent research shows that domestic smart meter data can support the DNO by increasing the visibility and reliability of the LV network [324], several limitations restrict the acquisition of domestic smart meter data: (1) Referring to the BEIS technical specification [12], the maximum interval resolution of the domestic smart meter is limited to 10 s, which cannot be used to detect transient faults in the distribution network and track the

[1] Referring to ANSI C84.1-1989 and IEE 141-1993 standard, voltages range between 50 V and 11kV volts are LV, 120, 220, 230 volts are the most common LV values used in the domestic, commercial, and industrial applications; voltages range between 1 kV and 100 kV are classified as MV; and voltages lower than 345 kV but higher than 100 kV are HV.

intermittent distributed renewable generations. (2) Referring to the data access regulation of OFGEM [41] and BEIS [12], the DNO cannot access the individual's smart meter data directly without any pre-processing methods for privacy consideration. Hence, a feeder-level smart meter at the LV distribution network between the MV/LV transformer and the domestic smart meter is proposed to fill the knowledge gap in the existing smart metering system.

4.4.3.1.1 Advanced sensor and meter at distribution/feeder level

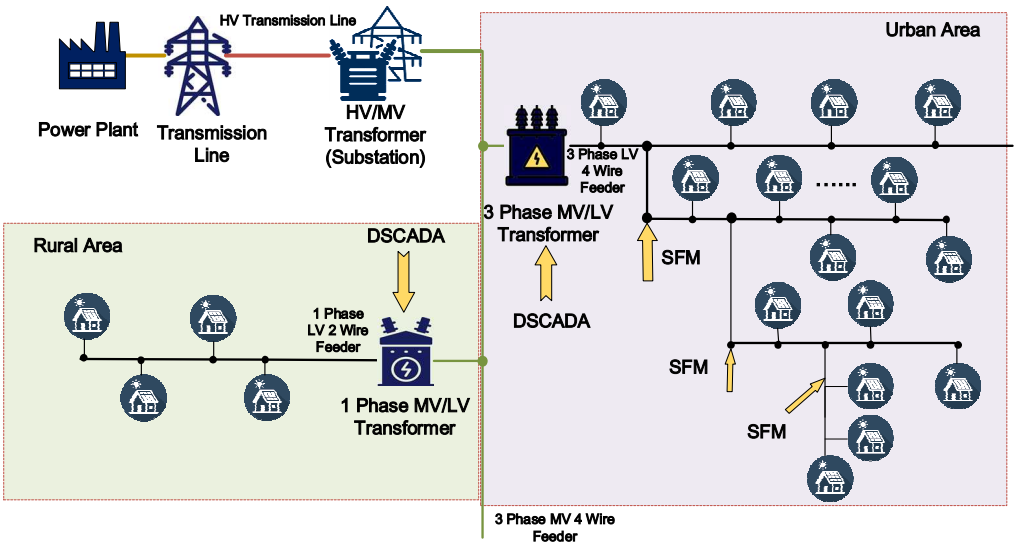
The term smart meter is a broad definition beyond the meaning of the domestic smart meter. There are multi-levels of smart measurement and data acquisition devices in the smart grid and the smart metering system, from the transmission network to the distribution network. This subsection introduces a hierarchical metering structure at the distribution level. The metering structure contains Distribution SCADA (DSCADA), Smart Feeder Meter (SFM), and a domestic smart meter.

1) **SFM:** Similar to a domestic smart meter, SFM is an advanced metering device that enables two-way communication and real-time access to the electricity parameters. The difference between SFM and a domestic smart meter is that the SFM is installed at the distribution feeder's head, and SFM can measure the network electrical parameters, e.g., active and reactive power of the demand load, the power factor, RMS values of line and phase voltages, line and phase currents. SFM has a higher sampling frequency without the regulation of BEIS (In [206], and SFM with an interval resolution of 1 minute is employed to obtain network measurements).
2) **Distribution SCADA (DSCADA)**: DSCADA is the SCADA system used for monitoring and controlling the distribution network (voltage level below 35 kV), it enables the interface with the data from SFMs, and the DSCADA can monitor the entire distribution system with reliable and secured operations [325].

4.4.3.1.2 Flexible multi-level physical aggregation scheme

Based on the three different levels of smart metering/ sensing devices: domestic smart meter, SFM, and DSCADA, a hierarchical physical aggregation scheme is proposed, see Figure 4-5 (a). Whilst the domestic smart meter only collects elasticity consumption from a single house, the SFM measures the electricity parameters of a distribution feeder, combining several residential houses. As for DSCADA, it collects the measurements from all SFM to monitor the overall distribution network. Referring to Figure 4-5 (b), a DSCADA system is installed at MV/LV transformer, multi-level SFMs are installed on the distribution feeders, and the aggregation size depends on the number of houses under the feeder. By drawing from a set of flexibly aggregated measurements, the DNO can operate and manage the distribution network without accessing domestic smart meters, using this hierarchical metering infrastructure.

(a) Hierarchical metering infrastructure to enable physical aggregation.



(b) Topology of a distribution network.

Figure 4-5. A low-voltage distribution network topology and a flexible multi-level physical aggregation scheme.

Table 4-4 presents 24 typical standard feeder models developed by GridLAB-D's feeder taxonomy [326]. The capacity of the feeder models ranges from 948 kW to

17021 kW, which indicates light rural areas to moderate urban areas. The approximate house number under each feeder model is estimated by fitting house-level power consumption data from Dataport [327]. From the table, it is observed that in a light rural area, only 205 houses are supplied by the feeder, while a feeder in a moderate urban area, more than 3600 houses are linked with the feeder.

The advantage of the physical aggregation scheme can be concluded as follows:

1) The feeder-level smart sensors/meters are not under the regulation of BEIS specification; hence, a higher sampling frequency can be adopted to improve the future distribution network's stability and reliability and enable the DNO to manage the distributed renewable energy better.
2) The feeder-level smart sensors/meters measure the aggregated parameters of a regional area (a street or a block) without invading personal energy consumption data, so privacy is guaranteed.

Table 4-4. Summary of prototypical feeders [326].

Feeder Model	kV	kW	Approximate Houses Number Under the Feeder	Description
R1-12.47-1	12.5	7152	1552	Moderate suburban
R1-12.47-2	12.47	2836	615	Moderate suburban
R1-12.47-3	12.47	1362	295	Small urban centre
R1-12.47-4	12.47	5334	1157	Heavy suburban
R1-25.00-1	24.9	2105	457	Light rural
R2-12.47-1	12.47	6046	1311	Light urban
R2-12.47-2	12.47	6098	952	Moderate suburban
R2-12.47-3	12.47	1411	344	Light suburban
R2-25.00-1	24.9	17021	3692	Moderate urban
R2-35.00-1	34.5	8893	1929	Light rural
R3-12.47-1	12.47	8417	1826	Heavy urban
R3-12.47-2	12.47	4322	937	Moderate urban
R3-12.47-3	12.47	7880	1230	Heavy suburban
R4-12.47-1	13.8	5530	1199	Heavy urban
R4-12.47-2	12.5	2218	481	Light suburban
R4-25.00-1	24.9	948	205	Light rural
R5-12.47-1	13.8	9430	2045	Heavy suburban
R5-12.47-2	12.47	4500	976	Moderate suburban
R5-12.47-3	13.8	9200	1996	Moderate rural
R5-12.47-4	12.47	7700	1670	Moderate suburban
R5-12.47-5	12.47	8700	1887	Moderate suburban
R5-25.00-1	22.9	12050	2613	Heavy suburban
R5-35.00-1	34.5	11800	2560	Moderate suburban
GC-12.47-1	12.47	5200	1127	Single large commercial

4.4.3.2 Informatic aggregation approach

Another aggregation method named the informatic aggregation approach (see Figure 4-6), which constructs a data aggregator AGG to collect α neighbouring smart meter readings (P_{SM_1}(t), P_{SM_2}(t),…, P_{SM_α}(t)) via LAN, AGG sums up the readings from all meter and only send the aggregated reading $f_{\mathcal{P}}^{agg}(t)$ to the energy utility referring to Equation (4-1). The aggregated reading is then transmitted to the energy utility and DCC via WAN, and the data aggregator is operated by a trusted entity such as DNO to avoid additional information leakage. The advantage of the informatic aggregation scheme is that the aggregation size can be controlled more flexible than the physical aggregation scheme, and the aggregation process is not limited to the geographic position of the smart meters.

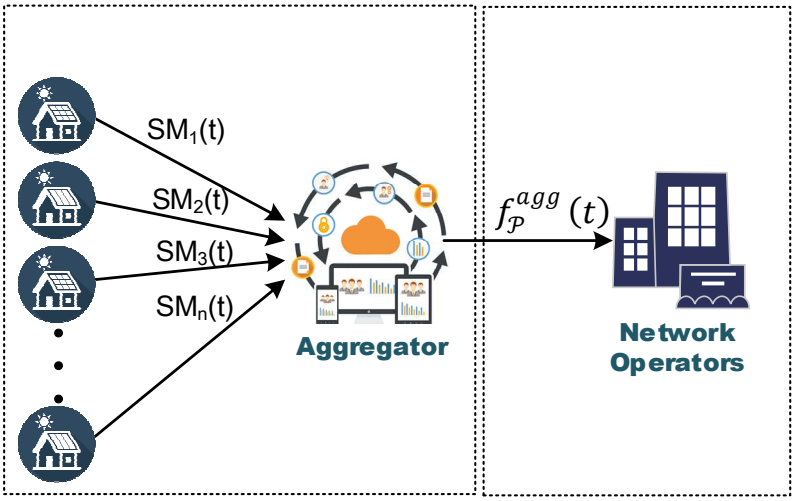


Figure 4-6. Informatic aggregation scheme via Local Area Network.

4.4.4 Time-of-use billing channel

The accuracy and trustworthiness of the consumption profile and the bills is the most critical target of the smart metering system. Although the smart meter can gather real-time power parameters, only the active energy consumption (measured in kilowatt-hour (kWh)) during a period is needed for billing purposes. The proposed TOU channel enables the dynamic TOU tariff and guarantees the correctness of the bills. The real-time power consumption data in this channel will not be sent to the ES

directly. Instead, the TOU price is sent to the smart meter from the ES, and the active energy consumption is stored in pairs with the corresponding TOU price locally. At the end of each reporting period, the cumulative active energy consumption and the bills are generated from the stored data. The detailed TOU billing process is demonstrated as follows.

Step 1: **Initialization**. Denote D as the reporting period (days), d as the index of days, N as the total number of tariff price record points per day, n as the index of record points per day.

Step 2: Data generation and storage. The ES sends the tariff price π to the smart meter N times a day; the smart meter receives the tariff and generates a vector of tariff price π_d in day d: $\boldsymbol{\pi_d} = \{\pi_{d,1}, \pi_{d,2}, \cdots, \pi_{d,N}\}$, $\pi_{d,n}$ represents the tariff price in the interval n. Moreover, a corresponding vector of the energy consumption E_d is generated based on the measurement: $\boldsymbol{E_d} = \{E_{d,1}, E_{d,2}, \cdots, E_{d,N}\}$. The smart meter stores $\boldsymbol{\pi_d}$ with $\boldsymbol{E_d}$ in pairs every day. At the end of each reporting period D, the energy consumption matrix $\boldsymbol{E}$ and tariff matrix $\boldsymbol{\Pi}$ are generated:

$$\boldsymbol{E} = \begin{bmatrix} E_{1,1} & E_{1,2} & \cdots & E_{1,N-1} & E_{1,N} \\ E_{2,1} & E_{2,2} & & E_{2,N-1} & E_{2,N} \\ \vdots & & \ddots & \vdots & \\ E_{D-1,1} & E_{D-1,2} & \cdots & E_{D-1,N-1} & E_{D-1,N} \\ E_{D,1} & E_{D,2} & & E_{D,N-1} & E_{D,N} \end{bmatrix} \tag{4-2}$$

$$\boldsymbol{\Pi} = \begin{bmatrix} \pi_{1,1} & \pi_{1,2} & \cdots & \pi_{1,N-1} & \pi_{1,N} \\ \pi_{2,1} & \pi_{2,2} & & \pi_{2,N-1} & \pi_{2,N} \\ \vdots & & \ddots & \vdots & \\ \pi_{D-1,1} & \pi_{D-1,2} & \cdots & \pi_{D-1,N-1} & \pi_{D-1,N} \\ \pi_{D,1} & \pi_{D,2} & & \pi_{D,N-1} & \pi_{D,N} \end{bmatrix} \tag{4-3}$$

Step 2: **Billing calculation**. Referring to matrix $\boldsymbol{E}$ and $\boldsymbol{\Pi}$, The total bills in £ B_{total} and the total energy consumption E_{total} $(kW \cdot h)$ during D are computed as:

$$E_{total} = E_{total} + E_{most\_recent\_value} \tag{4-4}$$

$$B_{total} = B_{total} + \pi_{most\_recent\_value} E_{most\_recent\_value} \tag{4-5}$$

Instead of sending detailed $\boldsymbol{E}$ and **Π,** only E_{total} and B_{total} is sent to ES, the ES then assigns a bill to the consumers.

Step 3. **Re-initialization**. After each reporting period, energy consumption and tariff price records are eliminated.

The storage capacity required: Suppose $N = 15\ minute$, $D = 30\ days$, and the data is recorded in the format of CSV, the data parameters recorded include the energy consumption, corresponding tariff price, and the date and the UNIX timestamp. The storage capacity required per day is 2.03 KB, and the total storage capacity required for one reporting period is 60.9 KB. Referring to the DBEIS specification, the storage capacity of the existing smart meter already satisfies the storage requirement.

4.4.5 Value-added service channel

As defined in the threat/adversary model in Section 4.2, the third party is the honest-but-curious adversary motivated to detect personal information from the shared data. Referring to the European Commission [328], the value-added service should be optional; the use and collection of data and by who needs to be specified as well as the specific purpose and where the data will be stored should be strictly identified. In the proposed smart metering system, a value-added service channel is designed as an optional choice for the consumers; two schemes localized platform and the federated learning-based cloud platform, are introduced in Chapter 5. The main idea of the channel is to prevent personal information from being shared with third parties via WAN, and the local model is trained inside the house to support data analysis locally.

4.5 Privacy Boundary of the Proposed System

The information flow diagram of the proposed smart metering system is shown in Figure 4-7. After mitigations implemented in Section 4.4, the external adversary still can eavesdrop on the communication of each channel. The information can be inferred from the high-frequency aggregation channel, and the TOU billing channel is

aggregated by reading $f_{\mathcal{P}}^{agg}(t)$, the cumulative energy consumption E_{total} and bills B_{total}. This section will investigate whether the adversary still infers sensitive information from the proposed system and to what extent the sensitive information can be hidden.

Although BEIS [12] states that the smart meter data after aggregation no longer belongs to personal information, recent works, however, have demonstrated that DNOs can infer individual household consumption data from feeder-level data, even though DNOs do not have permission to access individual smart meter data [74]. Moreover, although in the TOU billing channel, only the active energy consumption during the period is reported, recent research indicates that a short reporting period can also reveal personal information [329]. Based on the discussion above, two characteristics of the smart meter data are investigated: aggregation size α and interval resolution σ. A NILM-based data mining algorithm used by the adversary model is developed to demonstrate the adversarial process, and a three-level privacy boundary (real-time surveillance, presence/absence detection, complete protection) is presented.

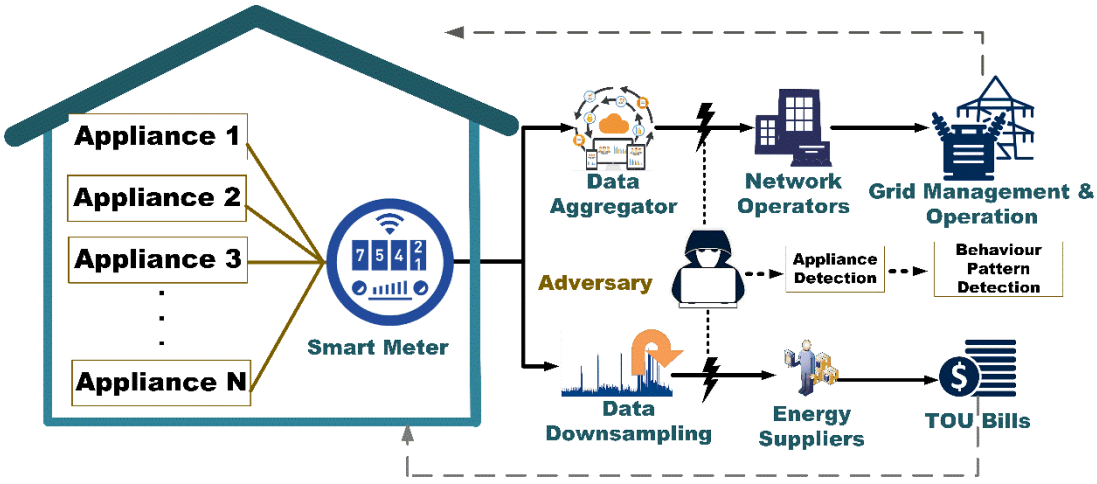


Figure 4-7. Information flow of the proposed system.

4.5.1 Data mining algorithm used by the adversary

After obtaining the smart meter data, the adversary $\mathcal{A}$ will try to infer private information such as detailed electricity activities of the individuals. By identifying all working electronic appliances inside the consumer's house, the consumer's privacy is

exposed to $\mathcal{A}$. NILM algorithm, as introduced in Chapter 2, is the potential data mining algorithm used by $\mathcal{A}$ to infer personal information from the smart meter data. NILM algorithm enables $\mathcal{A}$ to disaggregate the household power consumption curve into a series of appliance usage profiles, and $\mathcal{A}$ can further extract behaviour patterns of the individual of these appliance profiles [47]. The detailed disaggregation process of NILM is introduced as follows.

The power consumption recorded by the smart meter at the time slice is denoted as $t \in \mathcal{T} := \{1,2,\cdots,T\}$ as $P_{SM,t}$. $P_{SM,t}$ can be decomposed into individual appliance signals via the NILM algorithm:

$$P_{SM,t} = \sum_1^N Y_{i,t} \quad (i \in \{1,2 \cdots N\}, t \in \mathcal{T} := \{1,2,\cdots,T\}) \tag{4-6}$$

where $Y_{i,t}$ is the power consumption of electrical appliance i (range from 1 to N) at time slice t. Denoting the appliance profile sequence matrix as $Y^{N\times T}$:

$$Y^{N\times T} = \begin{bmatrix} Y_{1,1} & \cdots & Y_{1,T} \\ \vdots & \ddots & \vdots \\ Y_{N,1} & \cdots & Y_{N,T} \end{bmatrix} \tag{4-7}$$

In this chapter, a 1D CNN-LSTM NILM algorithm is employed as the data mining algorithm used by the adversary. Based on Equation (4-7), the deep neural network is constructed as below, and detailed hyperparameters settings can be found in Table 4-5.

Table 4-5. Data mining model settings.

Hyperparameters	**Value**	**Description**
Learning rate ϵ	0.05	The steps to minimise error.
Optimiser	Adam	
Number of LSTM/GRU layers	4	
LSTM/GRU units per RNN layer	512	
Number of 1D CNN layers	1	Extracting features from time-series data
kernel size of 1D CNN layer	5	The sliding window size of the 1D CNN
Batch size *B*	128	The number of training examples utilised in one iteration.
Activation function for hidden layers	ReLU	$f_{ReLU} = \max[0, z]$.
Activation function for the output layer	ReLU	Positive Output.
Epoch number	100	One cycle through the entire training dataset.
Loss function	MSE	Minimise the error between ground truth and prediction
Dropout	0.5	Reduce overfitting

The performance of NILM on a single house and original interval resolution achieves 83% accuracy on average [330], which convinces that the adversary has high computation ability in detecting behaviour patterns. The aggregation size α as well as down-sampling resolution σ are increasing steadily until the appliance usage information is not detected by NILM.

4.5.2 Implementation

4.5.2.1 Dataset construction

The data adopted in this chapter are The Reference Energy Disaggregation Data Set (REDD) [331] and Pecan Street Dataport (Dataport) [117], see Table 4-6. Both two datasets contain appliance-level and house-level power consumption data. Hence, the load profiles and appliance signatures can be obtained from the datasets. Nine typical household appliances are selected for this research, which is: air conditioner (AC), microwave oven (MO), electric vehicle (EV), water heater (WH), dishwasher (DW), dryer (DRY), stove (STO), furnace (FUR), refrigerator (REF). Three variables related to the appliance, the power rating, minimum duration, and power threshold, are described in Table 4-7. The power threshold in the table represents the minimum power to operate the appliance. The threshold is the minimum power to start the device; when the power is larger than the power threshold, the appliance is regarded as "on". Minimum duration represents the minimum operating hours of a particular appliance throughout the day. Furthermore, the rated power is the highest power input allowed through a particular device.

Table 4-6. Dataset description.

Dataset	Interval Resolution	NUM. of Houses	NUM. of Submeters	Duration
Dataport[117]	1 min	$\gg 1000$	75	4 years
REDD[331]	3 s	6	20	30 days

Aggregation Size Dataset: Referring to Section 4.4.3 and (4-1), Houses inside an aggregation group are picked randomly from two datasets to make up the new dataset. Then the new dataset is split into training/testing datasets (90% for training and 10% for testing). The input data of the model is the aggregated power consumption $f_{\mathcal{P}}^{agg}(t)$,

and the output of the model is the power consumption of a particular appliance $Y^{i,t}$ in house i.

Table 4-7. The property of appliances [117, 332, 333].

Appliance	Rating (kW)	Threshold (kW)	Min Drn (h)	Adversary Acc. (%)
Microwave Oven (MO)	1.5	0.30	0.025	0.77
Stove (STO)	1.2	0.24	2	0.89
Air Condition (AC)	2.0	0.40	12	0.85
Furnace (FUR)	1.0	0.20	8	0.91
Electric Vehicle (EV)	3.0	0.50	4	1.00
Refrigerator (REF)	0.055	0.01	24	0.94
Water Heater (WH)	3.5	1.00	2.5	0.75
Dryer (DRY)	2.1	0.7	1	0.76
Dishwasher (DW)	1.2	0.15	2	0.90

Interval Resolution Dataset: The new dataset is generated by reducing the interval resolution to σ, and then the new dataset is divided into training/testing datasets; both the input (household power consumption) and the output (appliance consumption) are from the same house i.

4.5.2.2 Privacy metrics for appliance detection

Once the adversary model is designed, the adversary's performance should be evaluated and quantified. In this section, two performance metrics that assess the performance of DNNs are introduced.

4.5.2.2.1 F-measure (F1 score)

F-measure is a performance measurement for classification adopted in NILM works and privacy measures [334, 335]; see Equation (3-22) in Chapter 3. Usually, when the F - measure is smaller than 0.5, the classifier is inadequate.

4.5.2.2.2 Correlation analysis

The Pearson correlation coefficient ρ is used to measure whether two continuous variables are linearly associated. The value of ρ ranges from -1 to 1 (a positive value indicates positive correlation, while a negative value indicates negative correlation);

the larger ρ, the stronger the correlation between two variables. The expression of the Pearson correlation coefficient is shown in Equation (4-8):

$$\rho = \frac{\sum_{t=1}^{n}(x_t-\bar{x})(y_t-\bar{y})}{\sqrt{\sum_{t=1}^{n}(x_t-\bar{x})^2 \sum_{t=1}^{n}(y_t-\bar{y})^2}} \quad (4\text{-}8)$$

where n is the sample size, x_t is appliance power consumption at time t and y_t power consumption generated by the adversary; $\bar{x}$, $\bar{y}$ is the mean value of x_t and y_t. A benchmark is presented for the following analysing process; see Table 4-8. An appliance is measurable when two metrics, F-measure and ρ, are lower than 0.2.

Table 4-8. Benchmarks of privacy metrics in appliance detection.

Performance	F-measure	Pearson correlation coefficient (ρ)
Poor privacy protection	0.5-1	0.5-1
Fine privacy	0.2-0.49	0.2-0.49
Good privacy	0.01-0.19	0.01-0.19
Perfect privacy	<0.01	<0.01

4.5.3 Results and discussion

This section quantifies the privacy boundary influenced by aggregation size α and interval resolution σ. Two case studies are designed for each parameter; the detectability of appliances and algorithms sensitivity in two privacy-preserving schemes are thoroughly investigated. A discussion based on the results is also presented to demonstrate the proposed three-level privacy benchmarks.

4.5.3.1 Privacy boundary level based on electrical events

Household appliances can be divided into three categories, loads depending on the characteristics and operating duration of the loads [336]. Detailed classifications are described as follows:

(1) Continuous load: A continuous load means that the device consumes energy throughout the day, such as the refrigerator and freezer, and the computer and printer in "standby" mode. Since residents' activities do not influence the continuous loads, these loads contain little sensitive information.

(2) Intermittent load: These appliances are not always on, but they are active enough to be recorded by the lowest hourly smart meters, such as air-conditioners, electric vehicles, furnaces, and water heaters.

(3) Active load: Power use appliances in an active house, such as Microwave oven, dishwasher, stove, and dryer.

Based on the load categories introduced above, three-level privacy boundaries are defined:

Level I (Real-Time Surveillance): All loads, including continuous loads, intermittent loads, and active loads, are detected by NILM. The adversary knows the entire life cycle of all residents (sleeping pattern, number of residents, when people leave their homes, etc.). Private information of residents is at high risk at this level.

Level II (Presence/Absence Detection): Both continuous and intermittent loads are detected by NILM. Under this privacy level, the adversary knows whether residents are inside/outside the house, but the adversary cannot monitor all electrical activities inside a house.

Level III (Complete Protection): No event is detected by the adversary, or only continuous loads are detected by NILM. Under this level of protection, the adversary cannot infer any sensitive information from given data.

4.5.3.2 Privacy boundary of aggregation size

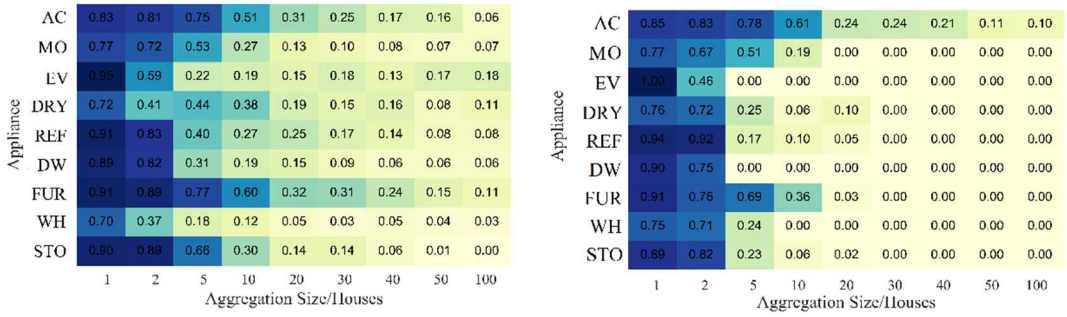
Recall $f_{\mathcal{P}}^{agg}(t)$ function in Equation (4-1), the aggregation size α is an essential variable that influences the statistical privacy. The purpose of NILM is to detect appliance usage; the precision of detection is evaluated when α is increasing steadily.

4.5.3.2.1 Detectability of appliances from aggregated data

$\mathcal{A}$ has high accuracy in appliance detection in a single house, raising privacy issues related to the smart meter. Recall the threshold identified in Table 4-7; an appliance is defined as detectable when the F-measure and ρ are both higher than 0.2. This case

study investigates nine typical appliances introduced in Table 4-7. These appliances represent continuous load (REF), intermittent load (AC, EV, WH, FUR), and activate load (MO, DW, STO, DRY), respectively. A 1D CNN-LSTM model with four LSTM layers is adopted as $f_{\mathcal{A}}(t)$. It achieves high efficiency in detecting appliances in a single house sees Table 4-7. By steadily increasing α from 1 to 100, the number of smart meters inside an aggregator is enlarged.

Figure 4-8 presents a heat map to show the performance of NILM in appliance detection given different α. As expected, the detectability of NILM is high with a small aggregation size ($\alpha < 5$). By continuously increasing α, both F-score and ρ decrease consequently, which means the appliance detectability is also reduced. Appliances such as EV, DW, and WH turn undetectable when α reaches 10. Most of these appliances operate during peak time, and load components under the aggregation scheme are extremely complex during this duration, so the inference process of NILM is easily blocked. As α reaches 20, MO, STO, DRY, and REF turn undetectable. It should be noted that Heating, Ventilation, and Air Conditioning (HVAC) devices such as AC and FUR remind detectable even $\alpha = 40$. The large α is that HVAC devices have a long operational duration (8-12 hours per day) and high-power rating (1-2 kW). To blind NILM for these HVAC devices, a minimum number of 50 houses are required. Figure 4-8 takes MO, DW, REF, and AC as examples to compare information inferred by $\mathcal{A}$ and the ground truth data under the aggregation scheme with $\alpha = 1, 2, 5, 50$, respectively.



(a) (b)

Figure 4-8. Heatmap of the performance of the NILM on appliances with different aggregation sizes (a) Pearson correlation coefficient (b) F-measure.

Since different appliances have different characteristic properties (Rating, Threshold, Minimum Duration), the performance of NILM on different appliances varies greatly. Based on the results shown in Figure 4-8, a correlation analysis is implemented between appliance characteristic properties and adversary detectability (shown in Table 4-9), and it is observed that the three characteristics almost show equal correlation with the adversary detectability (0.44 for Rating, 0.50 for Threshold, and 0.53 for Minimum Duration). To summarize, appliances with high ratings, high threshold, and long duration (such as AC, FUR, DRY) require larger α to blind NILM.

Table 4-9. Correlation between appliance characteristic properties and the detectability.

	Rating	Threshold	Minimum Duration
α	0.44	0.50	0.53
σ	0.34	0.26	0.74

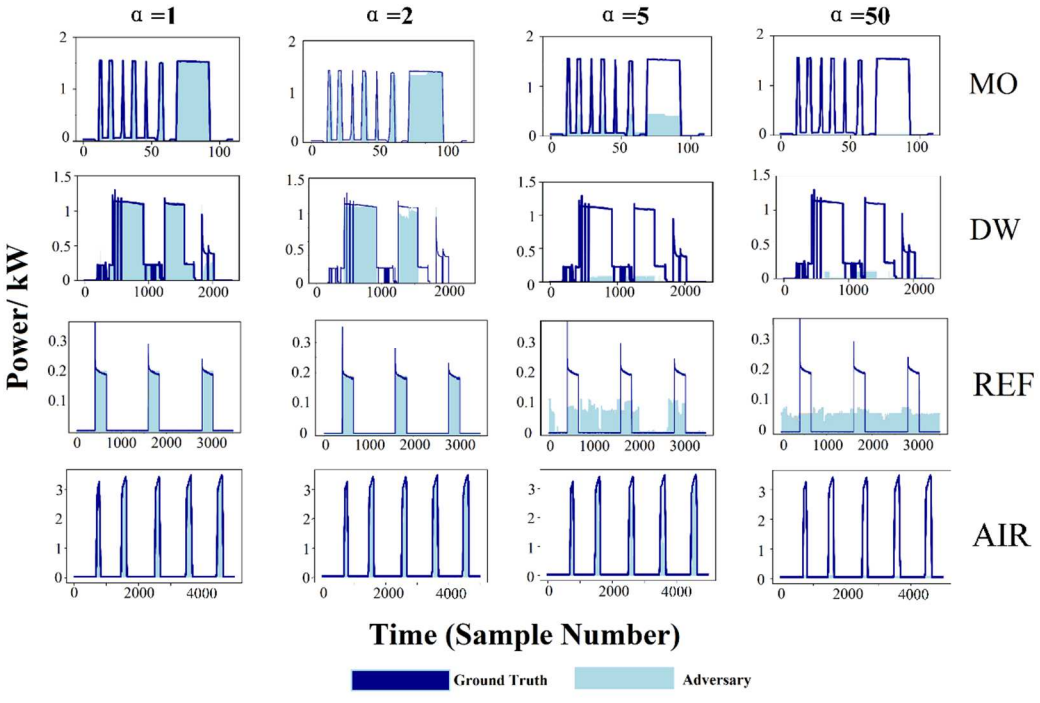


Figure 4-9. Examples of information inferred from the NILM and ground truth data in the aggregation scheme.

4.5.3.2.2 Sensitivity of algorithms of the aggregated data

Rather than the CNN-LSTM algorithm adopted in previous sections, the adversary can also adopt different deep learning-based NILM algorithms. This case study

discusses the sensitivity of algorithms in an aggregation scheme. Apart from the proposed algorithm, three state-of-the-art NILM algorithms are proposed, GRU [337, 338], CNN [131, 339], and the k-nearest neighbours (KNN) [340, 341] NILM algorithms are studied as well, referring to previous works. In Figure 4-10, each bar represents the average values of F-measure/ρ of all appliances with a particular algorithm. It can be found that all algorithms have desirable detectability on a single house (F-measure>0.77, and ρ>0.78), and CNN-LSTM has the best performance among all algorithms, followed by the GRU, while CNN and KNN have similar performance. The machine learning algorithm, KNN, is the most sensitive to the parameter α, as KNN-based NILM turns blind when α>10 while other three NILM models can still infer private information with high accuracy at this level. Moreover, CNN-LSTM and GRU have similar characteristics throughout the whole simulation, both CNN-LSTM-based NILM and GRU-based NILM lose general detectability when α>30 (it should be noticed that the general detectability only represents average privacy metrics of all appliances, some specific appliances are still detectable). To sum up, the proposed aggregation scheme for all algorithms discussed in this section, as the detectability of four is efficient algorithms drops to near zero at high aggregation size (α>40).

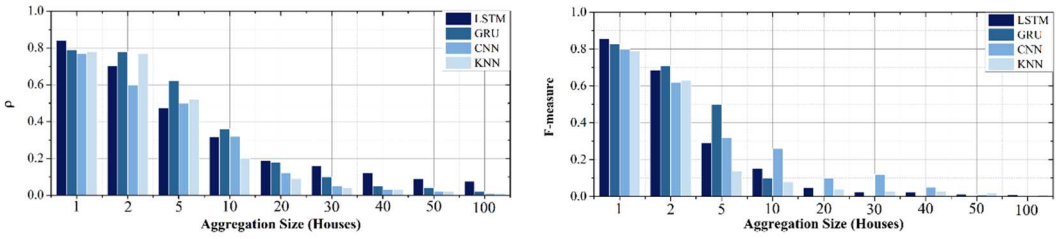


Figure 4-10. Comparison of different adversary algorithms in the aggregation scheme (a) Pearson correlation coefficient (b) F-measure.

4.5.3.3 Identifying the boundary of interval resolution

The privacy boundary of another critical parameter, interval resolution σ, is discussed in this section. Same as Subsection 4.5.3.2, two case studies are implemented to investigate the appliance detectability and algorithm sensitivity. The original interval

resolution of the dataset τ is 3 s; it should be noted that the sampling rate of the data in the current smart metering system is 15 minutes, and data with 3s is only available in experimental datasets.

4.5.3.3.1 Detectability of appliances from down-sampled data

In this subsection, the detectability of NILM on appliances regarding different σ is discussed. From the heatmap shown in Figure 4-11, all appliances are high detectable when σ<5 min except for MO. Appliances such as MO have a very high rating (1.5 kW), but the operation duration is short (0.025 hours). Hence when interval resolution increases, MO becomes challenging to be detected. As shown in Table 4-9, appliance detectability in the data down-sampling scheme correlates with a minimum duration (0.72) and is followed by Rating (0.34). Appliances with long operation duration require a significant σ value to hide sensitive information. For instance, AC requires at least one-hour interval resolution to blind NILM, and σ>5 h is required by EV. As for continuous load such as REF, which operates all day, σ should be larger than ten h. To summarise, σ>10 h is required to provide complete privacy. Figure 4-12 takes MO, DW, and REF as examples to compare information inferred by NILM and the ground truth data under a data down-sampling scheme with $\sigma = 3\ s, 5\ min, 0.5\ h, 2\ h$, respectively.

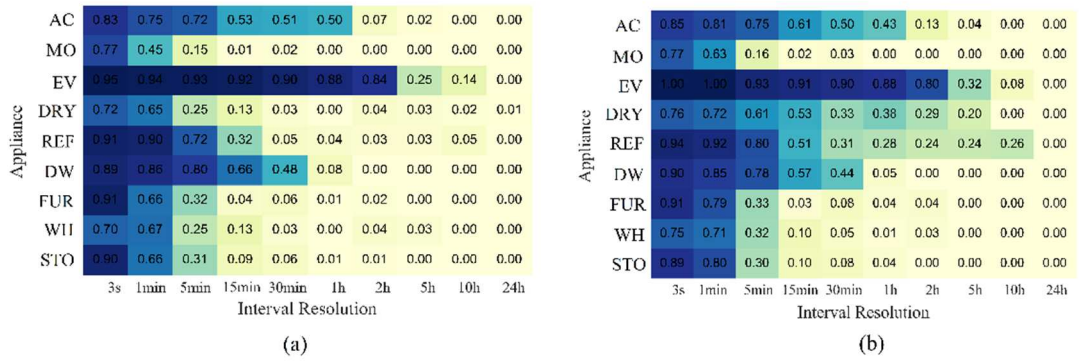


Figure 4-11. Performance of the NILM on appliances with different interval resolutions (a) Pearson correlation coefficient (b) F-measure.

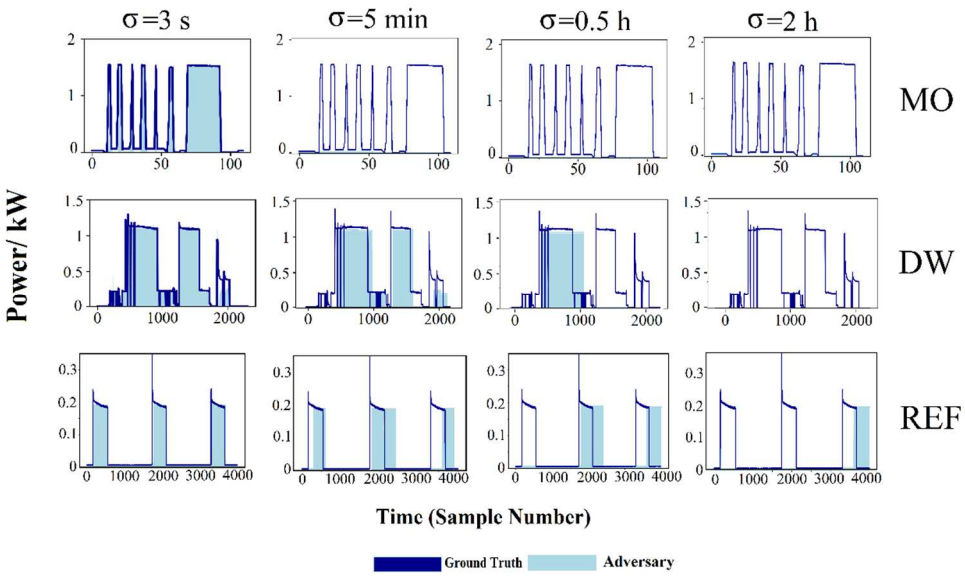


Figure 4-12. Examples of information inferred by NILM given different interval data.

4.5.3.3.2 Sensitivity of algorithms of the down-sampled data

Four adversaries with different algorithms (CNN-LSTM, GRU, CNN, KNN) are introduced to determine the algorithms' sensitivity in a data down-sampling scheme. As shown in Figure 4-13, the increase of σ weakens the detectability of all four adversaries significantly. It is essential to point out that all adversaries still maintain high inference ability when σ ranges from 15 to 30 min, while the sample frequencies of most smart meters in the UK are in this scope. A benchmark of σ=10 h is a safe threshold for the privacy-preserving model against the data mining algorithm used by the adversary.

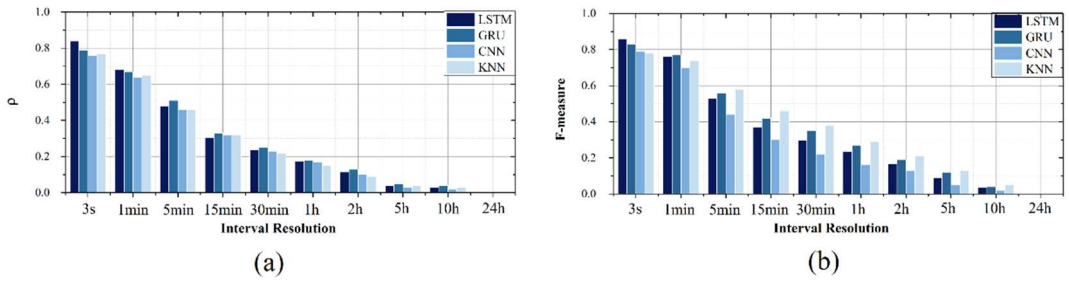
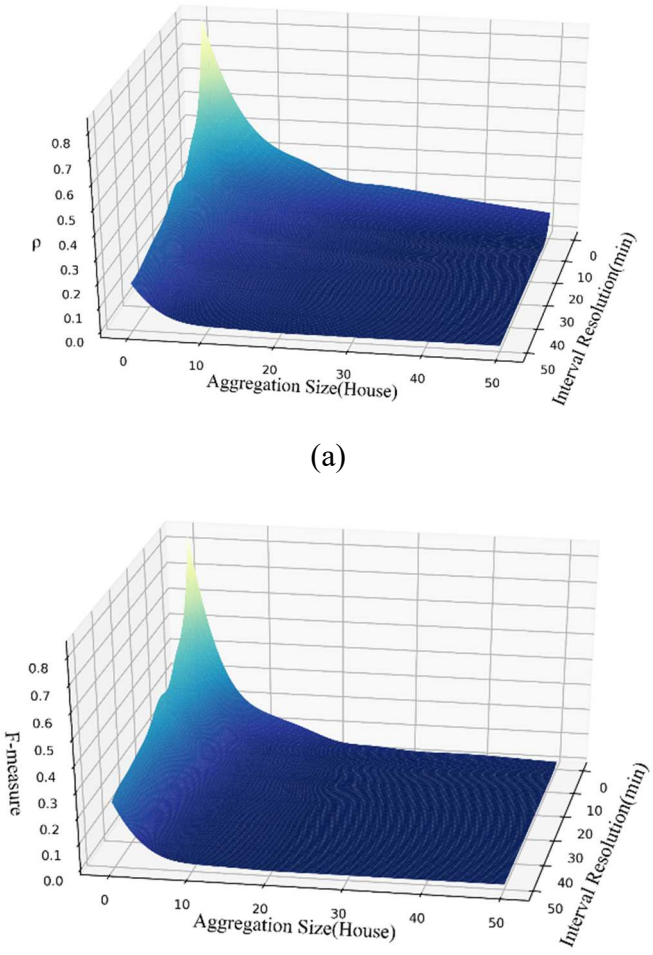


Figure 4-13. Performance of the NILM on appliance under different interval resolution (a) Pearson correlation coefficient (b) F-measure.

4.5.3.4 The combined effect of interval resolution and aggregation size

This section demonstrates the combined effect of two parameters, α, and σ, on the adversary computing ability. The aggregation size α and interval resolution σ are changed synchronously, and the dynamic variation of two privacy metrics, F-measure and ρ, is observed. The simulation results are presented in Figure 4-14, which uses 3D models to show dynamic changes. From the figure, it can be found that the detectability recedes rapidly, and both F-measure and ρ drop to zero given $\alpha > 10$ and $\sigma > 30\ min$.



(a)

(b)

Figure 4-14. 3D model of the privacy performance of the adversary with two parameters.

4.5.3.5 Determined privacy boundary

Based on simulation results and quantification of appliance detectability obtained in previous sections, three-level privacy boundaries are concluded in Table 4-10. When $\alpha < 20$ or $\sigma < 5\ h$, consumers are under privacy level I, which represents consumers under real-time surveillance at this level. By detecting appliance signatures of active loads (MO, DW, STO, and DRY), NILM can know detailed behaviour patterns of residents inside the house. When $20 \leq \alpha < 40$ or $5\ h \leq \sigma < 8\ h$, the consumers are under privacy level II, $\mathcal{A}$ can infer presence/absence information from intermittent loads (AC, EV, WH, FUR) but cannot understand complex behaviours inside the house. Finally, when $40 \leq \alpha$ or $8h \leq \sigma$, the consumers are under privacy level III; at this level, consumers are protected entirely and free of privacy concerns. In addition, when the Co-effects of two parameters are considered, the detectability of NILM drops dramatically compared to a single parameter; when $10 \leq \alpha$ and $30min \leq \sigma$, privacy level III is already achieved.

Table 4-10. Quantification of three-level privacy boundaries.

Privacy level	Appliance to detect	Quantification (Single parameter)	Quantification (Co-effects of two parameters)
Level I	MO, DW, STO, DRY	$\alpha < 20\ or\ \sigma < 5h$	$\alpha < 2\ and\ \sigma < 5min$
Level II	AC, EV, WH, FUR	$20 \leq \alpha < 40\ or\ 5h \leq \sigma < 8h$	$2 \leq \alpha < 10\ and\ 5min \leq \sigma < 30min$
Level III	All appliances	$40 \leq \alpha\ or\ 8h \leq \sigma$	$10 \leq \alpha\ and\ 30min \leq \sigma$

4.6 Privacy Risk Analysis

As introduced in the threat/adversary model in Section 4.2, both the internal and external adversary could infer sensitive information from the original smart metering system. In the proposed system, the following information is shared:

- The cumulative energy consumption E_{total} and bills B_{total} during the reporting period, D is sent to the ES.
- The aggregated smart meter data $f_{\mathcal{P}}^{agg}(t)$ from the aggregator to the DNO.

Referring to the privacy boundary evaluated in Section 4.5, the NILM data mining algorithm is blinded when aggregation size α reaches 40 houses, and the interval resolution exceeds eight-hour. Considering a multi-channel smart metering system with aggregation size $\alpha > 40$ houses and reporting period $D > 8h$, the following conclusion can be made:

1) An internal employee in ES and DNO cannot obtain personal data as 15-minute household smart meter data is never shared with ES and DNO.
2) The third-party service provider, the honest-but-curious adversary, cannot access the individual meter readings due to the access control (which will be demonstrated in detail in Chapter 5).
3) The external adversary who eavesdrops on the communication channel cannot infer personal data or sensitive information related to individuals, as the data mining algorithm they employ is blinded to the aggregated and down sampled data, as demonstrated in Section 4.5.

Based on the statement above, the privacy risk introduced by the threat/adversary is eliminated/reduced in the proposed multi-channel smart metering system.

4.7 Chapter Summary

This chapter presents a multi-channel smart metering scheme to prevent privacy risks raised by the smart meter. This chapter starts with identifying the threats/adversaries who bring privacy risks to the smart metering system. Then based on the defined threats/adversaries, GDPR, and the minimal data required by different entities, a smart metering system which can transmit different granularity data via three communication channels are developed. Based on the proposed system, functionalities include TOU billing, distribution network operation, and third-party value-added services. Moreover, a NILM-based data mining algorithm is introduced to quantify the privacy boundary (aggregation size and interval resolution) of the smart meter data transmitted in the proposed system. Finally, by comparing the privacy risks of the existing/proposed smart metering system, the conclusion is

reached that the proposed system can reduce the information leakage the threats/adversaries raised.

Chapter 5 Differentially Private Federated Learning-based Value-Added Service Platform

5.1 Introduction

5.1.1 Motivation and background

In the last chapter, a multi-channel smart metering system is introduced, enabling three communication channels between the consumers and other stakeholders (energy suppliers, distribution network operation, and third parties). Among all channels, the low-frequency TOU billing channel and high-frequency aggregation channel are the most vital channels to support compulsory functionalities, including TOU billing and grid operation and management, while the third-party communication channel is an optional choice to provide various value-added services to the consumers. Such services can introduce new market opportunities and engage the innovation of the electricity market [304], and smart meter crates opportunities to innovate in B2C and G2C projects (B2C – Business to Consumer, G2C – Government to Consumer) [342]. Various value-added services are available to consumers, including demand response, NILM, energy awareness and load forecasting. The software companies may also try to link their smart speakers (Echo [343], Google Home [344]) to the consumer's smart meter to help the consumers improve their energy awareness [342]. These value-added services to the consumers are based on characteristics of a household's energy consumption, while different services may require metering data with different resolutions.

For traditional value-added services in AMI, the power consumption data collected by smart meters are uploaded to a centralized server. The server can use the data to train machine learning/deep learning models, and then the trained model can make

predictions. However, these centralized value-added services and the data collected by the smart meter are subject to privacy concerns. As introduced in Chapter 4, the TP service provider is a potential internal adversary that would follow the protocol but try to harvest consumer data since it can have great commercial value [105]. Secondly, most value-added services require consumers to send detailed power consumption profiles of their houses or specific appliances with timestamps. Attacks such as NILM attacks [235, 345] can extract detailed behaviour patterns of consumers by disaggregating power consumption into detailed appliance usages. Thirdly, referring to data privacy legislation such as the European Commission's GDPR [301], data collected by the smart meter belongs to personal data, and the collection or storage of such information is strictly limited by the data minimization principle consent principle [346]. Moreover, the European Commission also suggested that value-added services should have separate communication channels where the type of data to be collected and stored should be specified [328].

The increasing popularity of smart meters has been accompanied by little attention being paid to these and other privacy issues on value-added services. Although several privacy-preserving methods could be used in different parts of AMIs, such as employing rechargeable batteries for smart meters [57], noise-adding methods [72], and data anonymization methods [85], there is still a need to target data collection by value-added services. Federated learning (FL) is a suitable technique to satisfy all suggestions proposed in [45]; this decentralized machine learning scheme enables clients to train local models without sharing private data with the server. Moreover, DP provides a stronger privacy guarantee when the cloud server collects model parameters from the clients [148].

5.1.2 Knowledge gap and limitation of existing work

Existing AMI mostly focuses on billing and monitoring services [135], and the smart meter only passively measures consumers' overall power consumption. In the next generation AMI, rather than a sensor, the smart meter plays the role of an edge device and the gateway of the smart home [347]; the smart meter is expected to implement

data analytics, prediction, and energy management with low communication latency. Moreover, the AMI need a private separate value-added service platform which can integrate third-party software and provide multiple value-added services [147].

In [45], M. Asghar et al. provided several suggestions and outlooks for future privacy-preserving value-added services. These suggestions can be concluded as follows: (1) implement value-added services on customers' private computing platforms (such as mobile phones and personal computers). (2) Develop new privacy-preserving distributed machine learning algorithms to provide better privacy guarantees to consumers.

In the literature for privacy-preserving in AMI, some developments are missing that can be concluded as follows.

1) Although some works discuss privacy-preserving value-added services, a hybrid platform that enables various services still needs to be redesigned to follow the GDPR strictly.
2) The existing smart metering system can only share 15-minute interval meter data with TP due to BEIS specifications, and only half-hourly data is stored. However, value-added services may require multi-resolution data, with data with intervals higher than 15 and 30 minutes.
3) Lack of work combines state-of-the-art privacy-preserving techniques (such as differential privacy and federated learning) with advanced deep learning methods (such as the attention-based deep neural networks).

5.1.3 Objective

The value-added service platform proposed in this chapter should satisfy the following requirements:

- As an honest-but-curious adversary, TP should follow the protocol to provide high-quality services to the consumers, while TP cannot access the individual's meter reading directly.
- The proposed platform should enable multi-resolution smart meters for various value-added services.
- The proposed platform should have an interface to receive information from other databases, e.g., metrological information from the weather station and solar irradiance information from the satellite.

5.1.4 Novelties and contributions of the chapter

Based on the knowledge gaps discussed above, the significant novelties can be summarized as follows.

1) A proposed privacy-preserving AMI TPS platform based on differential private federated learning (DPFL) scheme. The platform can provide multiple services to consumers without sharing their data (e.g., load demand data) to cloud servers and other parties.
2) An Attention Bidirectional Long Short-Term Memory (ATT-BLSTM) algorithm, one of the newest RNN models, is utilized as the local/central model to train the data and make predictions.
3) K-means clustering is used to cluster the clients into the normal and malicious clients using the local model weights only.

5.1.5 Structure of the chapter

The remainder of the chapter is organized as follows: Section 5.2 describes the preliminary knowledge and techniques used in this chapter, including differentially private federated learning and attention-based bidirectional long short-term memory. Section 5.3 illustrates the proposed value-added services platform and the training process of both the local and global models. In Section 5.4, the performance of the proposed model is evaluated, and several variables that influence the model accuracy

are fully investigated. The last section concludes the chapter and discusses future works.

5.2 The Preliminaries

This section introduces preliminaries of the proposed value-added service platform, including attention-based bidirectional long short-term memory recurrent neural network, differential privacy, and federated learning.

5.2.1 Attention-Based Bidirectional Long Short-Term Memory Recurrent Neural Network

Uncertainty, nonstationary, nonlinearity and long-term dependence are the time-series demand load data characteristics. RNNs are utilized to process the data to some extent. However, one of the drawbacks of RNNs is the long-range dependency problem [348], the capability of RNNs to process long sequence data is inefficient, and even long short-term memory (LSTM) turns forgetful in special cases. The attention mechanism is a probability weighting mechanism that was first proposed in 2014 [349]. ATT-BLSTM architecture improves its accuracy by assigning the probability weights to each previous hidden state to find the most informative for the output at the current time step [350] (Figure 5-1). Hence, the utilization of the attention mechanism can improve the output of the bidirectional LSTM (BLSTM) and better solve the long-term memory problem [350]. ATT-BLSTM model consists of two parts: the conventional BLSTM and an attention layer, see Figure 5-1. In a BLSTM structure, given a minibatch input $\boldsymbol{X}_t \in \Re^{n\times d}$ (n is the number of examples, and d is the sequence size of each example), the forward hidden state $\overrightarrow{\boldsymbol{h}_t} \in \Re^{n\times h}$ and backward hidden state $\overleftarrow{\boldsymbol{h}_t} \in \Re^{n\times h}$ (h denotes the number of hidden units) at time step t can be expressed as (5-1) and (5-2):

$$\overrightarrow{\boldsymbol{h}_t} = \phi(\boldsymbol{X}_t \boldsymbol{W}_{xh}^{(f)} + \overrightarrow{\boldsymbol{h}}_{t-1} \boldsymbol{W}_{hh}^{(f)} + \boldsymbol{b}_h^{(f)}) \tag{5-1}$$

$$\overleftarrow{\boldsymbol{h}_t} = \phi(\boldsymbol{X}_t \boldsymbol{W}_{xh}^{(b)} + \overleftarrow{\boldsymbol{h}}_{t-1} \boldsymbol{W}_{hh}^{(b)} + \boldsymbol{b}_h^{(b)}) \tag{5-2}$$

where $\boldsymbol{W}_{xh}^{(f)}, \boldsymbol{W}_{xh}^{(b)} \in \Re^{d \times h}$ and $\boldsymbol{W}_{hh}^{(f)}, \boldsymbol{W}_{hh}^{(b)} \in \Re^{h \times h}$ represent the weights of the model, and $\boldsymbol{b}_h^{(f)}$ and $\boldsymbol{b}_h^{(b)} \in \Re^{1 \times h}$ are the biases of the model. Then, by integrating the forward and backward hidden states, the hidden state is obtained as $\boldsymbol{h}_t \in \Re^{n \times 2h}$. Finally, H_t is fed to the output layer to compute the output $\boldsymbol{O}_\mathrm{t} \in \Re^{n \times q}$ (q is the number of outputs):

$$\boldsymbol{h}_\mathrm{t} = \left[\overrightarrow{\boldsymbol{h}_t}; \overleftarrow{\boldsymbol{h}_t}\right]^T \tag{5-3}$$

$$\boldsymbol{O}_\mathrm{t} = \boldsymbol{h}_t \boldsymbol{W}_{hq} + \boldsymbol{b}_q \tag{5-4}$$

where $W_{hq} \in \Re^{2h \times q}$ is the weight and $b_q \in \Re^{1 \times q}$ is the bias of the output layer. As for the attention layer, denoting the current hidden state as $\boldsymbol{h}_t$ and the previous hidden state as $\boldsymbol{h}_i$ ($1 \leq i < t$). Referring to the definition in [349], a context vector c_t is computed, which is the weighted sum of all hidden states:

$$\boldsymbol{c_t} = \sum_{i=1}^{t-1} \alpha_{t,i} \, \boldsymbol{h_i} \tag{5-5}$$

where $\alpha_{t,i}$ is the weight for the hidden state $\boldsymbol{h_i}$ at timestep t. An attention matrix $\alpha_{t,i}$ is obtained by adopting the softmax function, as shown in (5-6) and (5-7):

$$\boldsymbol{\alpha_t} = [\alpha_{t,1}, \alpha_{t,2}, \ldots, \alpha_{t,(t-1)}] \tag{5-6}$$

$$\alpha_{t,i} = \frac{\exp\,(e_{t,i})}{\sum_{k=1}^{T} \exp\,(e_{t,k})} \tag{5-7}$$

In the above equations, $e_{t,i}$ represents the score (or energy) of a feedforward neural network (denoted as function a), and the purpose of $e_{t,i}$ is to capture the influence of the previous hidden state $\boldsymbol{h}_i$ on the current hidden state $\boldsymbol{h}_t$. Three a functions are introduced in [351]: location-based attention function (*location*), general attention function (*general*), and concatenation-based attention function (*concat*) [349]. Detailed functions are illustrated below:

$$e_{t,i} = a(e_{t,k}) = \begin{cases} \boldsymbol{W}_e^{\top}\boldsymbol{h}_i + b_e & Location \\ \boldsymbol{h}_t^{\top}\boldsymbol{W}_e\boldsymbol{h}_i & General \\ \boldsymbol{v}_e^{\top}\tanh(\boldsymbol{W}_e[\boldsymbol{h}_t;\boldsymbol{h}_i]) & Concat \end{cases} \tag{5-8}$$

where $\boldsymbol{v}_e$ is the parameter to be learned by the neural network. Referring to the experiment implemented by [352], attention-based BLSTM achieves excellent performance in processing power consumption data as its characteristic in allocating the importance to the overall power consumption data points that correspond to the state changes of appliances. As a result, the model can better extract relevant features from the collected data.

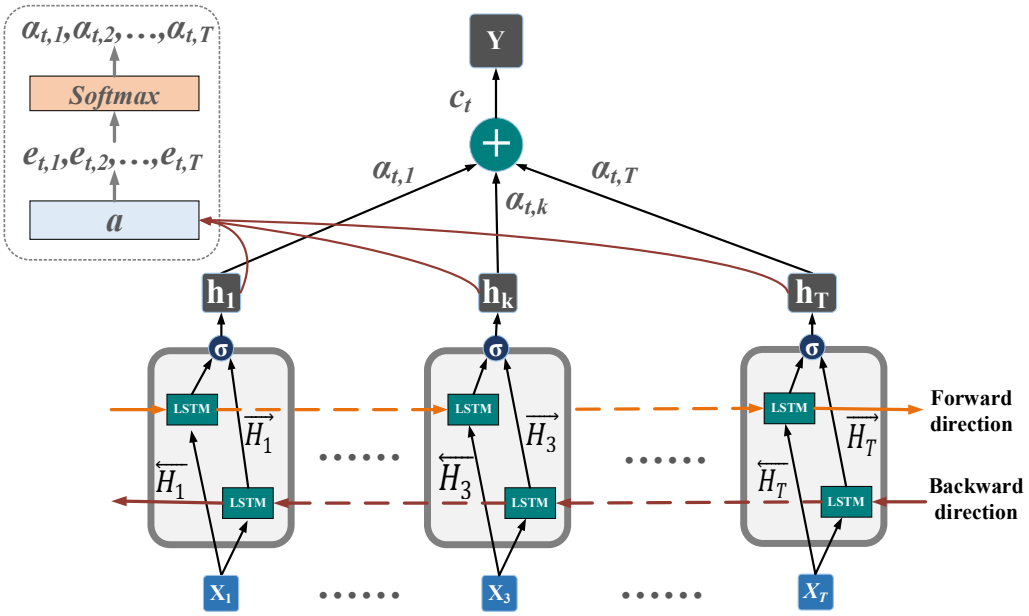


Figure 5-1. Structure of attention-based bidirectional LSTM.

5.2.2 Differential Privacy

Differential privacy is a technology proposed by C. Dwork in 2006 to protect an individual's identification information by adding random noise over the original aggregated data so that every individual has little effect on the result [84, 353, 354]. In this case, the adversary cannot distinguish the change of the aggregated data with/without one individual data point. Several noise addition mechanisms are available in the literature [354], including the Laplace, exponential, and Gaussian

mechanisms. The privacy level, ε, is guaranteed via the above noise addition mechanism, and the lower ε is, the higher the privacy level that can be achieved.

Definition 1. $\mathfrak{R}$ is a random function that transforms input β to a random output $\mathfrak{R}(\beta)$.

Definition 2. $d(\beta, \beta')$, which is the distance between two neighbouring datasets, represents the minimum number of individual samples required to shift dataset β to β'.

Definition 3. For a random function f, the global sensitivity, S_f, is the maximum difference between the outputs of two neighbouring datasets β and β'. S_f also determines the overall noise to be added into the DP mechanism.

$$\Delta f = \max_{d(\beta,\beta')=1} \|f(\beta) - f(\beta')\| \tag{5-9}$$

Definition 4. The Gaussian privacy mechanism denoted $\mathfrak{R}$ is defined as f plus the noise term $\mathcal{N}$.

$$\mathfrak{R}(\beta) \triangleq f(\beta) + \mathcal{N}(0, \Delta f^2 \sigma^2) \tag{5-10}$$

where $\mathcal{N}$ is the Gaussian distribution with mean 0 and standard deviation ${S_f}^2\sigma^2$.
The scale σ is computed as

$$\sigma = \sqrt{2\ln\left(\frac{1.25}{\delta}\right)}\Delta_2/\varepsilon \tag{5-11}$$

Definition 5. A randomized function $\mathfrak{R}$ satisfies (ε, δ) privacy $\mathbb{P}_{\mathbb{R}}$ for any two neighbouring datasets β and β':

$$\mathbb{P}_{\mathbb{R}}[\mathfrak{R}(\beta) \in \varepsilon] \leq e^{\varepsilon}\mathbb{P}_{\mathbb{R}}[\mathfrak{R}(\beta') \in \varepsilon] + \delta \tag{5-12}$$

where ε denotes all possible outcomes in range $\mathfrak{R}$, and δ is the possibility that the differential privacy is broken. In this work, 10^{-5} is selected as δ.

The following composition theorem computes the overall privacy cost throughout the learning process:

Theorem 1. (Composition Theorem) If f is (ε_1, δ_1)-differential privacy and g is (ε_1, δ_2)-differential privacy, then

$$f(D), g(D) \text{ is } (\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)\text{-Differential Privacy} \tag{5-13}$$

With the composition theorem, the overall privacy cost is calculated by accumulating the privacy cost at each training step. Hence, the overall privacy cost after T steps is:

$$\varepsilon_{total} = T\varepsilon; \delta_{total} = T\delta \tag{5-14}$$

5.2.3 Federated Learning with Differential Privacy

Federated learning is a decentralized machine learning algorithm that shifts the learning process from the centralized cloud server to decentralized clients [148]. An FL model contains $K \in \mathcal{N}^*$clients indexed by k and one cloud server denoted as S. The target of the FL algorithm is to minimize a local objective function that can be expressed as:

$$\min_{w \in \mathbb{R}^d} \frac{1}{m} \sum_{i=1}^{m} f_i(w) \tag{5-15}$$

For client $k \in K$, a local model will be trained with their private data on an edge device (such as a smartphone or laptop):

$$\forall k, w_{t+1}^k \leftarrow w_t - \eta \nabla \mathcal{L}(w_t) \tag{5-16}$$

The parameters of the local model w_{t+1}^k for a client are then sent to S, the parameters of all local models are aggregated, and a data-weighted average over all parameters is performed to update the global model w_{t+1}:

$$w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k \tag{5-17}$$

where n_k is the number of samples of client k, and n is the number of samples of all clients. Then, the new global model is broadcast to clients, and clients will retrain the local model with their data. The above steps will be repeated until convergence.

Although federated learning models avoid sharing private data with a cloud server or third parties, privacy is still a significant concern. By continuously sharing the parameters of local models, the adversary can still infer some sensitive information from the parameters [355]. DPFL provides a strong privacy guarantee and simultaneously reduces communication costs [356]. Hence, a DPFL algorithm is adopted in this work to provide a stronger privacy guarantee to the system. The DPFL adopted in this work is based on the randomized Gaussian mechanism introduced in [357]. Denoting the global model at timestep t as w_t; the model is optimized by the local model of client k, and the optimized parameters are denoted as w^k. The mismatch between w_t and w^k is client k's update and can be expressed as:

$$\triangle w^k = w^k - w_t \tag{5-18}$$

To reduce the sensitivity of $\triangle w^k$ with a considerable value, a scaling function is applied to $\triangle w^k$ to ensure that the second norm $\|\triangle w^k\|_2$ is limited by sensitivity S. Hence, the scaled version of the updates is obtained as:

$$\triangle \bar{w}^k = \triangle w^k / max\left(1, \frac{\zeta^k}{S}\right) \tag{5-19}$$

where $\zeta^k = \|\triangle w^k\|_2$ and S is the median of norms of clients' updates and can be expressed as:

$$S = median\{\zeta^k\} \tag{5-20}$$

By adding random Gaussian noise scaled to S, $\mathcal{N}(0, S^2 \cdot \sigma^2)$ into the sum of all scaled updates from K clients $\sum_{k=1}^{K} \triangle \bar{w}^k$, the Gaussian mechanism approximating the sum of updates is obtained. The new global model w_{t+1} is computed by adding the original global model with averaged approximation:

$$w_{t+1} \leftarrow w_t + \frac{1}{K}\left(\sum_{k=1}^{K} \triangle \bar{w}^k + \mathcal{N}(0, S^2 \cdot \sigma^2)\right) \tag{5-21}$$

5.3 Value-added service platform

This section introduces a privacy-preserving value-added service framework in AMI based on differential private federated learning. To simplify the system, the following assumptions are employed for the rest of the chapter: (1) the sampling frequency, computation ability, and types of data of all smart meters are the same; (2) latency and communication delay are neglected, and (3) all clients upload the parameters at the same pace.

There are two methods to enable edge computation on a smart meter, which are the Trusted Platform Module (TPM) introduced by [358] and a private platform (such as a mobile phone and computer). As for TPM, an extra computation module should be plugged into the smart meter to enable edge computation. The TPM can store cryptographic keys, and for performing cryptographic primitives using the keys, so the smart meter with a TPM is considered fully trusted [45]. The capacity of TPM is listed in Table 5-1. The microcontroller for TPM introduced in [145] is a Raspberry Pi 3, and this is a tiny and cheap (10 pounds per chip [359]) device which contains Central Processing Units (CPUs), Graphics Processing Units (GPUs) to support machine learning software such as TensorFlow and TensorFlow Lite. The advantage of TPM is that it reduces the possibility of being prone to viruses and malware as TPM disconnects from the Internet. Moreover, the power of TPM is supplied by the power unit of the smart meter; hence it is unlikely to lose connectivity due to the battery power. The limitation of TPM is that the method requires an extra module on the smart meter, which will increase the cost.

Another approach is using the high-computational personal device to perform the edge computation and train the local model. The smartphone and computer have been employed as edge devices for federated learning applications [360]. Examples of the hardware specification of the personal devices are listed in Table 5-1. It should be noticed that the disconnection of a small group of edge devices will not influence the performance of the federated learning model, as the global model is still being updated

with the rest of edge devices since the smartphone or personal computers are prone to viruses and malware, which would introduce potential security risks. Hence, the following security requirements are provided to guarantee mobile security.

1) The third-party APP should be restricted READ-ONLY access to the smart meter record, and the accessed data cannot be stored in the memory.
2) Consumers are required to install antivirus software to look for suspicious behaviours.
3) Strong and up-to-date cryptographic protocols must be employed to guarantee the confidentiality of the data transmitted between the client app and the backend server.
4) Block all the interaction from other mobile applications.
5) A firewall, intrusion detection system and recovery system are required to be set up to ensure the security of the backend server from cybersecurity attacks.

Table 5-1. Examples of hardware specifications for edge computing.

Edge Device	Product	GPU	CPU	Software Support
TPM [145]	Raspberry Pi 3	VideoCore VC6	Quad Core 1.2GHz Broadcom BCM2837 64bit CPU	TensorFlow and TensorFlow Lite
Smartphone	iPhone 13	4-core GPU 16-core Neural Engine	A15 Bionic chip	iOS 13
Computer [360]	NVIDIA Jetson TX2	NVIDIA Pascal	Dual Denver 2 64-bit + quad ARM A57	TensorFlow and Caffe

There are two value-added service platforms developed in this section, which are a localized service platform (benchmark model) and a federated learning-based platform; detailed description of the two models is presented as follows.

5.3.1 Benchmark model - Localized service platform

The localized platform follows the data minimization principle, preventing personal data from “leaving” consumers’ houses. Rather than sending personal data to the server of TPs, TPs send algorithms and models to TPM. The consumer then enquires from the local model to obtain wanted services. As shown in Figure 5-2, the process of the additional third-party service channel consists of the following steps, and all steps can be divided into two categories depending on the network (WAN or HAN):

Operations via WAN:

1) **Data Loading**: The training data is loaded from the public electricity database. The data is pre-processed (data cleaning, encoding, and feature scaling) before feeding into the global model.
2) **Model Training**: The global model is trained with the training data.
3) **Model Broadcasting**: After the global model is trained, the global model parameters are broadcasted to consumers.

Operations via HAN:

4) **Local Model Generation**: TPM downloads the global model parameters via WIFI and generates a local model with given parameters.
5) **Private Data Transmission**: The smart meter communicates with TPM and reports the power consumption data.
6) **Query Process**: The consumer can send a query to TPM, one the platform receives the query, it will evaluate the local model with private electricity data to compute the outputs of the query. Then a detailed report is sent to the consumer via IHD.

The data flow in Figure 5-2 shows that the consumer's electricity data are shared inside HAN and are never sent to the utility, but the services are enabled. The enabled services include NILM, STLF, and demand response.

However, the localized scheme also has several drawbacks that cannot be overlooked:

- By sending algorithms/model parameters to consumers, the model's parameters and training dataset would be stolen by users, while these models and datasets are confidential.
- Moreover, it is difficult to implement a privacy-preserving algorithm in complex models such as machine learning/deep learning-based services. Furthermore, the global model cannot be updated frequently to personalise to different customers.

Based on the limitations discussed above, A federated learning-based platform is developed.

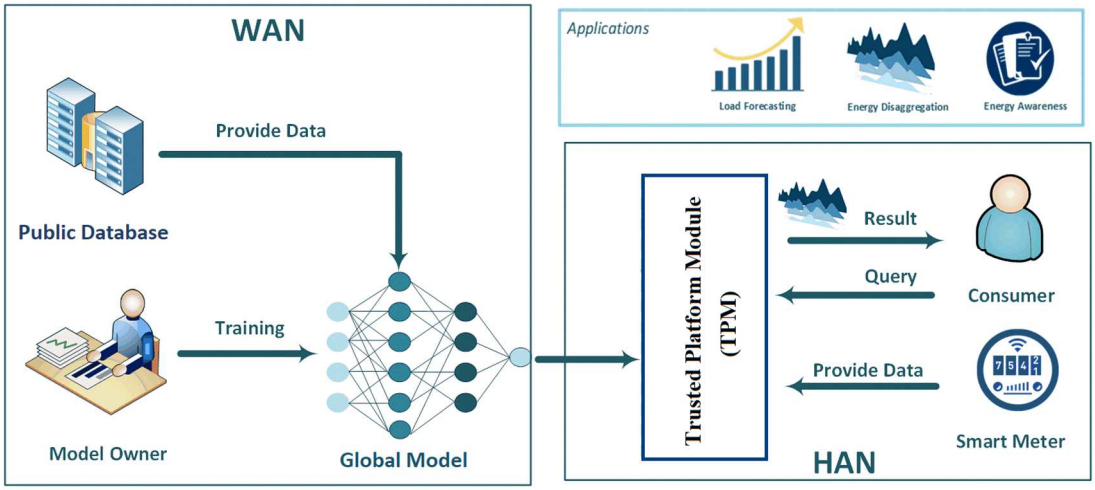


Figure 5-2. Privacy-preserving third-party service channel.

5.3.2 Federated learning service platform

The overall system is demonstrated in the flowchart shown in Figure 5-3. The clients in this framework are the consumers who install smart meters at home; they use IoT devices such as smartphones and personal computers to train local models and communicate with the cloud server. The proposed framework contains six procedures that can be concluded as follows:

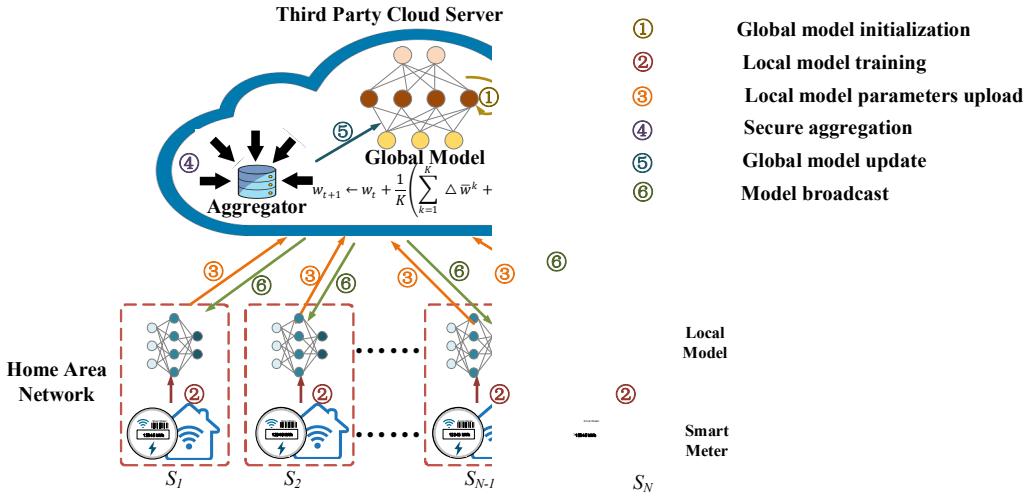


Figure 5-3. Overall differential private federated third-party service scheme.

- **Procedure 1. Global model initialization**. Initially, the global model at the TP cloud server is initialized by allocating random values to its parameters. Then, the model parameters are downloaded by clients and broadcast to local models.
- **Procedure 2. Local model training.** After receiving the parameters from the cloud server, the local model is updated in the IoT device; then, the IoT device will train the new model with private data locally.
- **Procedure 3. Local model parameters upload.** After the training process, the parameters of all local models are uploaded to the cloud server.
- **Procedure 4. Aggregation with differential privacy.** An aggregator is responsible for secure aggregation once it receives a response from the required number of clients. It aggregates the data with a random mechanism to maintain client-level differential privacy. After the aggregation of each round, the collected local model parameters are discarded.
- **Procedure 5. Global model update.** The global model is updated with the output of the aggregator.
- **Procedure 6. Model broadcast.** Parameters of the new global model are broadcast to all local models that run on IoT devices.

5.3.3 Local Deep Neural Network Model

As shown in Figure 5-4, the structure of the local neural network consists of seven layers:

- **The input layer:** The power consumption data collected by the smart meter are fed into the model.
- **Two BLSTM layers:** BLSTM is adopted to extract high-level representation from the input data. Although more BLSTM layers enable the model to better extract nonlinear features from the input sequences, too many BLSTM layers will cause overfitting problems, and the training time is also highly extended. Considering the above issues, two BLSTM layers are easier to implement with high efficiency.

- **An attention layer:** As introduced in Section 5.2.1, the attention layer utilizes the attention mechanism to rank the importance of the previous hidden states and selects the most informative hidden state to predict the output values.
- **A concatenated layer:** As the optional layer, the function of the concatenated layer is to load data from external databases related to evaluating the desired value-added service. External databases include meteorological, calendar, and electricity market databases.
- **A fully connected layer:** The fully connected layer links the recurrent layers with the output layer. The layer's purpose is to fully extract the nonlinear correlation between all input variables and outputs.
- **The output layer:** For classification tasks, the probability of each category is generated as the output; for regression tasks (such as load forecasting or NILM), the output layer generates the prediction value at the current timestep.

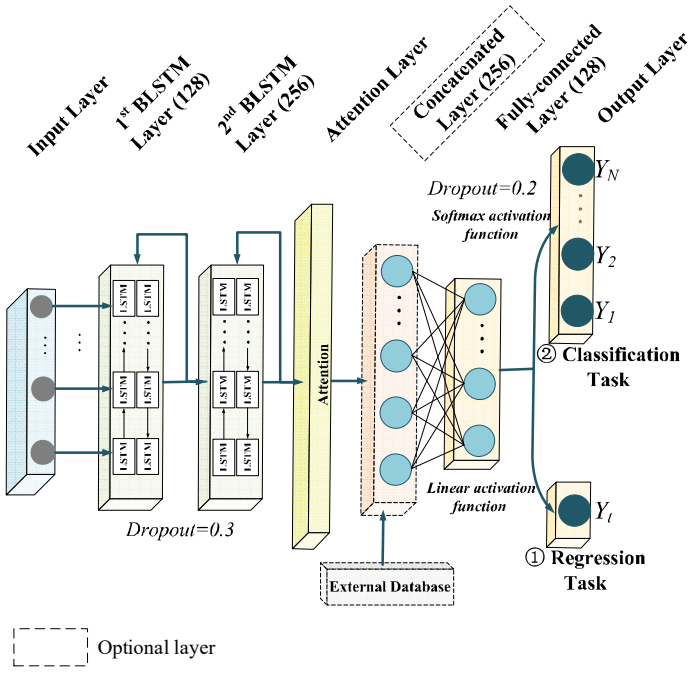


Figure 5-4. Structure of local neural network model.

5.3.4 Cloud Server

The central cloud server is responsible for secure aggregation and central model updates. Once the server receives the uploaded local models from all clients in each

communication round, it will implement a secure aggregation with differential privacy. As introduced in Section 5.2.3, random Gaussian noise is added to the sum of the clipped updates. Then, the aggregated updates are utilized to update the global model on the server. See Algorithm 5-1.

Algorithm 5-1: Differential Private Federated Learning-based Value-Added Service. communication round t; the maximum communication round T; the maximum pre-train communication round T_p; B is the mini-batch size; q is the fraction of clients; ε is the target differential privacy; σ is the Gaussian Mechanism parameter; δ represents the probability that ε-DP is broken, and Q is the threshold for δ.

Procedure DPFL(K, w_t)
  initialize the global model w_0 ▷ initialize weights of the global model on the server
  initialize Accountant (ε, K) ▷ initialize the privacy accountant on the server
  while $r < R$ **do**
    $\delta \leftarrow$ Accountant (ε, q) ▷ accumulate the privacy loss
    if $\delta > Q$ **then** return w_t ▷ return the model when the privacy threshold reached
    for client k in qK **do**
      $\triangle w_{t+1}^k, \zeta^k \leftarrow$ ClientUpdate(k, w_t) ▷ the client k's update and norm update on the local model
    $S = median\{\zeta^k\}$ ▷ compute the median norms of clients' updates as sensitivity
    $w_{t+1} \leftarrow w_t + \frac{1}{m}\left(\sum_{k=1}^{K} \triangle w_{t+1}^k / max\left(1, \frac{\zeta^k}{S}\right) + \mathcal{N}(0, S^2 \cdot \sigma^2)\right)$ ▷ update the global model
  return w_{t+1}
Procedure ClientUpdate(k, w_t) ▷ perform on client k
  $w \leftarrow w_t$
  while $r < r_{max}$ **do**
    for $b \in$ B **do**
      $w \leftarrow w - \eta \nabla \mathcal{L}(w_t)$ ▷ mini-batch gradient descent
    $\triangle w_{t+1} = w^k - w_t$ ▷ client k's local model update
    $\zeta = \|\triangle w_{t+1}\|_2$ ▷ second norm update
  return $\triangle w_{t+1}, \zeta$
Procedure K-MeansClustering $(X, \triangle w)$
  random place centroids C_1, C_2 across $\triangle w$
  repeat
    for i in K **do**
      $\gamma_{ij} = \begin{cases} 1 \text{ if } j = \text{argmin}_j \|\triangle w_i - C_j\|^2 \\ 0 \text{ otherwise} \end{cases}$ ▷ find the nearest cluster j for model i
    for j in 2 **do**
      $n_j = \sum_{i=1}^{K} \gamma_{ij}$ ▷ assign the data points to clusters
      $C_j = \frac{1}{n_j} \sum_{i=1}^{K} \gamma_{ij} \triangle w_i$ ▷ assign the average of points to cluster j
  until Convergence
  return C_1, C_2 ▷ assign the regular clients to C_1 and the malicious clients to C_2

5.4 Case study and discussion

In this section, the accuracy and efficiency of the proposed Clustered-DPFL Attention-BLSTM TPS framework are validated by using the scheme for a typical TPS residential STLF task. The proposed scheme and the traditional centralized framework are tested with real-world datasets. Moreover, the impacts of exogenous

meteorological and calendar features are also investigated. Finally, the privacy performance, as well as the communication cost, is studied as well.

5.4.1 Data description

In this chapter, a real-world dataset from Pecan Street Dataport [327] was used to evaluate the forecasting performance. The dataset contained over 1200 houses and was collected in Austin, Texas, the United States (N 30° 15', W 97° 43') between January 1st and December 31st, 2018. Both household and appliance power consumption in each house was recorded with sampling frequencies of 1 min and 15 min, respectively. This work selected 15 min interval smart meter data from 50 houses as the simulation dataset. The dataset was split into training data (1st January 2018 to 30th September 2018) and testing data (1st October 2018 to 31st December 2018). The training data were split into 36-week data, and one-week data were adopted for each communication round. When the communication round reaches 36, it will start dragging data from the first week again at the next communication round until it reaches the threshold of δ. Moreover, meteorological data resources from the same location (Austin, Texas, US) are used; the data is provided by National Centres for Environmental Information (NCEI) [193].

5.4.2 Implementation

5.4.2.1 Simulation environment

The case study is implemented on a workstation with a Core i7-7700HQ CPU, NVIDIA GTX 1060 GPU (8 cores), and 8GB RAM. The DPFL ATT-BLSTM is operated on Python 3.6 with Pytorch [361], and the privacy loss is computed via the Tensorflow-Privacy library [362].

5.4.2.2 Evaluation metrics

The performance of the scheme is evaluated with Normalized Mean Absolute Error (nMAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error

(RMSE). The smaller value of MAE, MAPE, and RMSE, the better performance the model provides.

5.4.2.3 Benchmark model

Several benchmark models are designed better to demonstrate the accuracy and robustness of the proposed method. Firstly, the proposed model is compared with three different service frameworks, such as centralized framework, localized framework, as well as FL framework, without adding noise during the aggregation process:

(1) Conventional centralized ATT-BLSTM model (denote as Centralized model). In the Centralized model, the DNN algorithm only runs on the cloud server, and the server will collect the power consumption data from all connected smart meters. The collected data is then used for training the centralized DNN model. Finally, the server will send the trained model back to the consumers.
(2) FL ATT-BLSTM model without DP (denote as FL model). The structure of the FL model is precisely the same as the proposed DPFL model; the only difference is that no noise is added during the aggregation process.
(3) Localized ATT-BLSTM model (denote as Localized model). The smart meter can only train the DNN model with minimal data in the Localized model.

Then three benchmark models under the DPFL framework utilizing different DNN algorithms (MLP, LSTM, BLSTM) are selected. By comparing the proposed model with the models listed below, the efficiency of ATT-BLSTM can be validated.

(4) DPFL model utilizes LSTM as a training algorithm (denote as DPFL-LSTM model).
(5) DPFL model utilizes BLSTM as a training algorithm (denote as DPFL-BLSTM model).
(6) DPFL model utilizes MLP as a training algorithm (denoted as DPFL-MLP model).

5.4.3 Hyperparameters configuration

The hyperparameters of the pre-training model and the proposed Clustered-DPFL Attention-BLSTM model are summarized in Table 5-2. The pre-training model is a shallow MLP with only one dense layer. The layer contains 16 cells, and the activation function of the dense layer is the Rectified Linear Unit (ReLU), which enables the model to learn nonlinear correlations better. The optimizer is SGD with the learning rate 1×10^{-3}.

Figure 5-4 and Table 5-2 show that the DPFL ATT-BLSTM model contains two BLSTM layers, with 128 and 256 cells, respectively. Followed by an attention layer with a size of 28 and one dense layer with 128 cells. The activation function of hidden layers is ReLU, and the optimizer is Adam with the learning rate 1×10^{-4}. As the STLF task is a regression task, the output layer size is one. Moreover, dropout and L2 regularization are used to avoid overfitting problems. 0.3 and 0.2 are selected as the dropout rates of the BLSTM layer and the dense layer, respectively. And 1×10^{-3} is selected as the weight decay value.

Table 5-2. Hyperparameter configuration.

Pre-training model	
Hyperparameter	Value/range
Layers	1 Fully connected layer with 16 cells
Batch size	32
Activation function	ReLU
Epochs	3
Optimizer	SGD
Learning rate	1e-3
Dropout rate	0.3
Differential privacy federated learning model	
Hyperparameter	Value/range
Lookback	4
Optimizer	Adam
Loss	MSE
Activation function	ReLU
Layers	2 BLSTM layers with 128 and 256 cells, respectively; 1
Epochs for each client in every communication round	5
Privacy budget ε	1, 2, 4, 6, 8, 10, 12
δ	1e-1, 1e-2, 1e-3, 1e-4, 1e-5, 1e-6, 1e-7, 1e-8
The GM parameter σ	1.12
Number of batches per client B	128
Dropout rate	0.5
Weight decay	1e-3
Attention size	28
Learning rate	1e-4
Total clients	5, 10, 50
Percentage of clients selected each round q	30%

5.4.4 Computation complexity

Computation complexity is evaluated in terms of the overall runtime of the service. To quantify the computation complexity, the following variables are defined:

- $Phase$ I: Implement the proposed DPFL model.
- $Phase$ II: Respond to the clients' query with the updated local model.
- $T_{local}(t)$: Time for clients to train local model at communication round t.
- $T_{agg}(t)$: Time for the central server to aggregate the local model parameters with differential privacy at communication round t.
- $T_{upload}(t)$: Time for the clients to upload the local models to the xth cluster server at communication round t.
- $T_{broadcast}(t)$: Time for the central server to broadcast the global model to the clients at communication round t.
- $T_{global}(t)$: Time for the central server to update the global model at communication round t.
- T_{qurey}: Time for the local server responses to the consumer's query.

In Phase I, also known as the federated learning period, all cluster servers are assumed to operate in parallel, so the runtime of the proposed model with client k in a communication round t can be estimated by the following equation:

$$T_{phase\mathrm{II}}(t) = T_{local}(t) + T_{upload}(t) + T_{agg}(t) + T_{broadcast}(t) + T_{global}(t) \quad (5\text{-}22)$$

Then the total time cost during Phase I is calculated as:

$$T_{phase\mathrm{II},total} = \sum_{t=1}^{T} T_{phase\mathrm{II}}(t) \quad (5\text{-}23)$$

Finally, the overall computation complexity of the proposed model is evaluated by

$$\begin{aligned} T_{total} &= T_{phase\mathrm{I},total} + T_{phase\mathrm{II},total} \\ &= \sum_{t=1}^{T} T_{phase\mathrm{II}}(t) + T_{qurey} \end{aligned} \quad (5\text{-}24)$$

5.4.5 Communication cost

During each communication round, the central server should communicate with its clients in the following steps: q percentage of overall clients K are selected as the training targets on each communication round. Hence there are a total number of qK clients who need to upload their local models to the central server. Hence, after T communication rounds, qKT communication sessions are needed to complete the whole training process.

5.4.6 Comparison of the proposed model with centralized and localized models

After filtering out the malicious clients, the federated model operates among all regular clients. In the first case study, the proposed DPFL scheme is compared with the conventional Centralized scheme, Localized scheme, and the normal FL scheme. To control the variable, all schemes utilize ATT-BLSTM as the DNN algorithm. The forecasting results are concluded in Table 5-3. Figure 5-5 plot the predicting load curves by the four schemes and the ground truth curve (solid blue line) in three consumers' houses. Considering the forecasting accuracy, the centralized scheme has the best performance, as the centralized scheme can access all consumers' data without any constraints. Accessing a more significant amount of the data will help the central model better learn the characteristics of the loads among all houses and avoid the overfitting problems, which will decrease the accuracy significantly. However, the centralized scheme suffers from significant privacy risks as all consumers must send their demand data continuously. The regular FL scheme almost achieves equal accuracy as the centralized scheme, especially when client number K increases. From Table 5-3, when $K = 50$, the nRMSE of the values forecasted by the FL scheme reaches 6.67%, which is only 2.33% less than the Centralized scheme. This simulation result confirms that the FL can achieve a similar forecasting performance as the Centralized scheme without sharing the real-measured data to the cloud server. In

other words, the FL scheme can satisfy the functionality requirement without scarifying individuals' privacy.

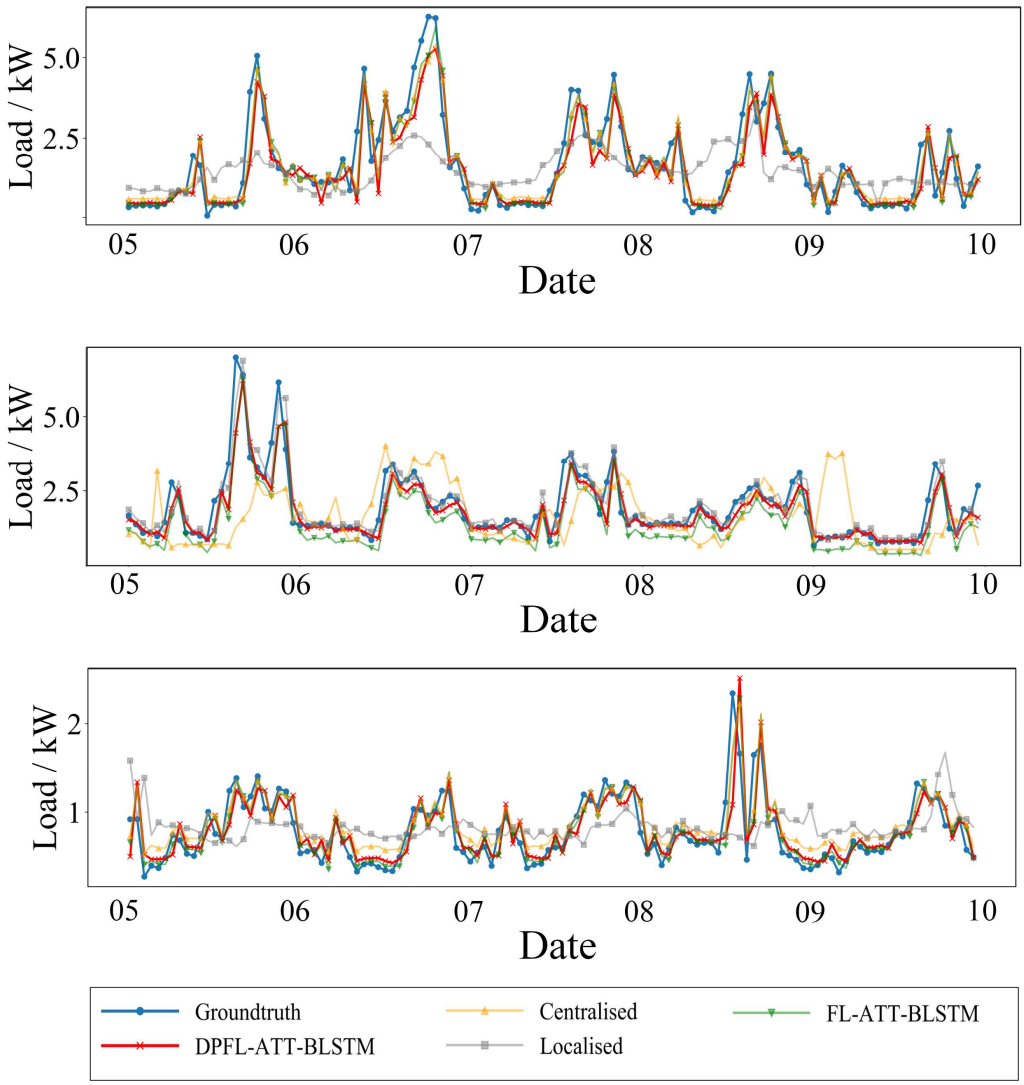


Figure 5-5. Short-term load forecasting results of three houses predicted by proposed differential private federated learning scheme and three conventional schemes (ε=8, δ=10^{-5}).

The Localized scheme disconnects communication with the cloud server, and all computation processes are completed within the smart meter and personal devices. From the predicted curve shown in Figure 5-5, the Localized scheme failed to predict the demand load in most situations. Also, the high nRMSE and nMAE errors presented in Table 5-3 convince the conclusion that the Localized scheme does not reach a balance between privacy and accuracy.

Table 5-3. Load forecasting performances of the proposed and benchmark models.

Model	ε	K	MAPE (%)	nMAE (%)	nRMSE (%)	C.R.	C.C.	CP.C (s)
DPFL-MLP	1	5	221.40	32.99	35.25	1	1	0.71
		10	99.60	26.52	28.97	1	3	0.69
		50	76.51	16.51	20.38	1	15	0.73
	4	5	78.67	25.16	26.16	6	6	13.50
		10	70.82	9.06	11.59	3	9	10.93
		50	70.41	7.68	10.99	3	45	41.33
	8	5	162.87	20.32	21.64	36	25	80.72
		10	69.58	8.08	10.85	15	45	88.04
		50	63.68	8.32	10.50	18	221	332.70
DPFL-LSTM	1	5	257.20	29.51	31.87	1	1	1.45
		10	146.04	15.08	19.39	1	3	1.56
		50	75.12	10.63	13.06	1	15	1.47
	4	5	94.83	14.14	17.41	6	6	30.17
		10	71.55	7.40	10.65	3	9	24.06
		50	71.43	11.61	13.30	3	45	94.08
	8	5	73.88	15.65	21.91	36	35	307.18
		10	68.31	7.57	10.97	15	45	345.70
		50	62.43	7.24	9.94	18	221	1422.23
DPFL-BLSTM	1	5	176.51	21.41	24.60	1	1	2.10
		10	152.10	12.13	17.11	1	3	2.26
		50	102.31	11.54	16.72	1	15	2.26
	4	5	79.17	15.46	16.49	6	6	45.89
		10	72.92	14.64	15.67	3	9	35.63
		50	70.98	9.72	12.13	3	45	150.98
	8	5	69.67	17.21	18.73	36	35	693.44
		10	65.95	10.01	11.68	15	45	718.48
		50	61.37	6.16	9.30	18	221	3159.73
DPFL ATT-BLSTM	1	5	323.89	16.27	20.44	1	1	3.29
		10	400.23	19.89	23.09	1	3	3.29
		50	376.45	29.65	41.20	1	15	4.98
	4	5	51.21	20.73	21.44	7	6	95.39
		10	40.38	7.13	10.53	3	6	42.81
		50	36.35	5.68	8.07	3	45	172.61
	8	5	29.06	4.49	8.04	36	35	307.22
		10	24.67	4.36	7.52	15	45	339.93
		50	14.44	4.32	6.92	18	221	1526.36
FL ATT-BLSTM	—	5	19.62	4.19	7.45	50	50	441.57
		10	17.20	3.76	6.70	50	147	1151.88
		50	12.59	3.70	6.67	50	735	4631.15
Centralised ATT-BLSTM	—	—	10.34	2.87	4.34	—	—	434.83
Localised ATT-BLSTM	—	—	68.73	10.01	10.69	—	—	29.21

C.R.: Communication round; **C.C.**: Communication cost; **CP.C**: Computation cost.

The DPFL scheme, the privacy-enhanced version of the normal FL scheme, can predict performance merely worse than the two schemes mentioned above. This is due to the privacy constraints set by DP. The privacy level of the DPFL scheme can be adjusted flexibly by setting the two DP parameters, the privacy budget ε and the probability of information being leaked δ. Typically, smaller ε means a smaller distance between the two neighbouring databases when sending a query. Hence the adversary has difficulty distinguishing these two databases by observing the query

output. Hence, a smaller ε provides better privacy but less accuracy simultaneously. From the results shown in Table 5-3, when ε=8 and δ=10^{-5}The DPFL scheme's performance is 3.75% and 12.31% worse than the FL scheme from the perspective of nRMSE and MAPE. Although the accuracy of the DPFL scheme stays below non-differentially private schemes, it is significantly better than the Localized scheme, which only trains the model with its data.

5.4.7 Comparison of the proposed model with other federated learning algorithms

In the first case study, the proposed DPFL ATT-BLSTM model is compared with DPFL models that utilize different DNN algorithms (benchmark models (4)-(5)). The forecasting results of all models are shown in Table 5-3, nMAE, nRMSE, and MAPE are used to measure the accuracy of the prediction results, and the communication cost, as well as computation cost, are recorded. The privacy budget ε range from 1 to 8, and the client number K range from 5 to 50. To visualize the performance of the proposed scheme and benchmark models, 30-minute forecasting results of random five houses are presented in Figure 5-6 (under the condition ε=8, δ=10^{-5}). In each communication round, only 30% of clients (e.g., 15 clients when $K = 50$) are selected to participate in the training process. Unlike feeder-level load forecasting, which has a regular peak load every day, household-level load forecasting is more challenging as the load profile on different days varies a lot. From the figure, DPFL-ATT-BLSTM performs best among all algorithms, and it is observed that the load curve predicted by the proposed DPFL-ATT-BLSTM model (solid red curve) tracks the ground truth load curve (solid blue curve) in most cases, both the peak part and the off-peak part are predicted with high accuracy. Considering the evaluation metrics, the proposed model has the lowest MAPE, nRMSE, and nMAE values in the same comparison group. Referring to the results shown in Table 5-3, when ε=8 and δ=10^{-5}, the nRMSE and nMAE value of the proposed model reduces by 31.95% and 11.22% compared to the DPFL-BLSTM.

Meanwhile, DPFL-MLP (light green solid curve) has the worst performance in most cases. Without the memory cell, it has very limited predictability in forecasting time-series data. From Figure 5-6, DPFL-MLP neither tracks the peak nor the off-peak load. However, this method also has an advantage: the computation cost is the least among all algorithms. When the computation ability of the edge devices is limited, this method could be the priority choice. DPFL-LSTM (solid orange curve) and DPFL-BLSTM (solid pink curve) models have similar prediction performances, while the DPFL-BLSTM model is slightly better. When ε=8 and $\delta=10^{-5}$, the nRMSE values of DPFL-LSTM and DPFL-BLSTM are 9.94% and 10.17%, respectively.

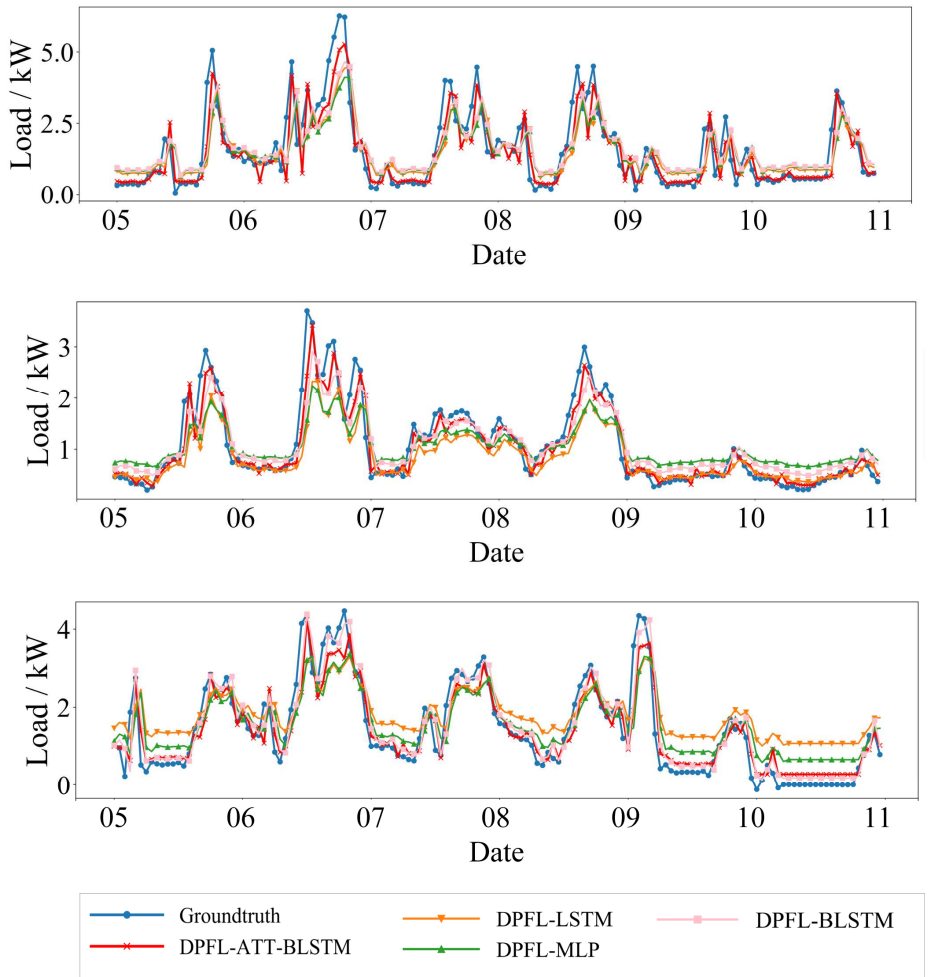


Figure 5-6. Short-term load forecasting results of five houses predicted by four differential private federated learning models (ε=8, $\delta=10^{-5}$).

These results demonstrate that the ATT-BLSTM is more efficient in processing time-series data, especially if the data is nonstationary and nonlinear. The ATT-BLSTM's

superior predicting performance can be summarized as follows: (1) The bidirectional structure enables the model to extract features from both forward and backward directions; (2) The attention mechanism helps the model find the essential hidden state of the current output.

5.4.8 Influence of client number

Another vital parameter that influences the performance of the proposed DPFL scheme is the client number K. Table 5-3 presents the model performance for $K \in [5, 10, 50]$. the privacy metrics, CRs, CC, and CPC for each case are also recorded. Referring to [363], The choice of DP parameter δ is influenced by K and should obeys the following constrain:

$$\delta \ll \frac{1}{K} \tag{5-25}$$

This condition avoids protecting most consumers' privacy by revealing a few consumers' [363]. According to this requirement, the threshold of δ, Q is set as 1×10^{-5}. From the Table 5-3, it is found that under the DPFL scheme, more clients achieve higher model accuracy: When $K = 5$, the prediction error is considerable high, and when K increases to 50, the accuracy of the model almost reaches the same accuracy as non-DP schemes. This is because more clients will reduce the additive noise's standard deviation during the secure aggregation process. Based on the above simulation results, the conclusion is that more clients can efficiently reduce the accuracy cost under the same privacy budget.

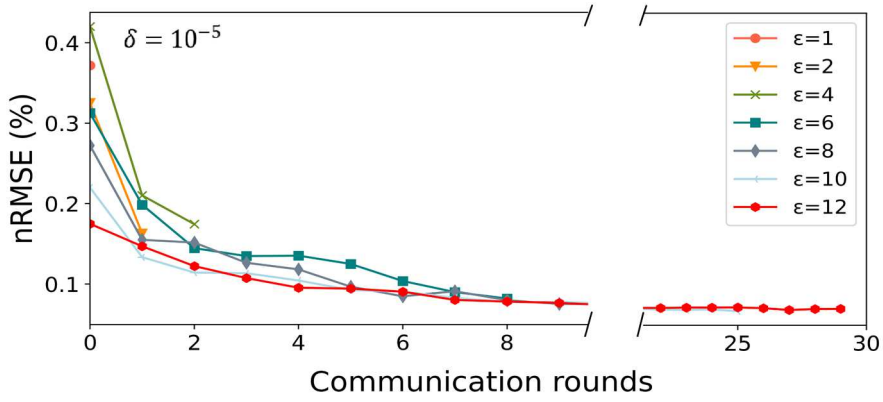
5.4.9 Influence of privacy budget

In the DPFL scheme, the most important parameters to make the trade-off between privacy and accuracy are the two DP parameters ε and δ. Recall Algorithm 5-1, during the secure aggregation process in each communication round, given ε and GM parameter σ; the central server accountant evaluates δ [357]. The central server will

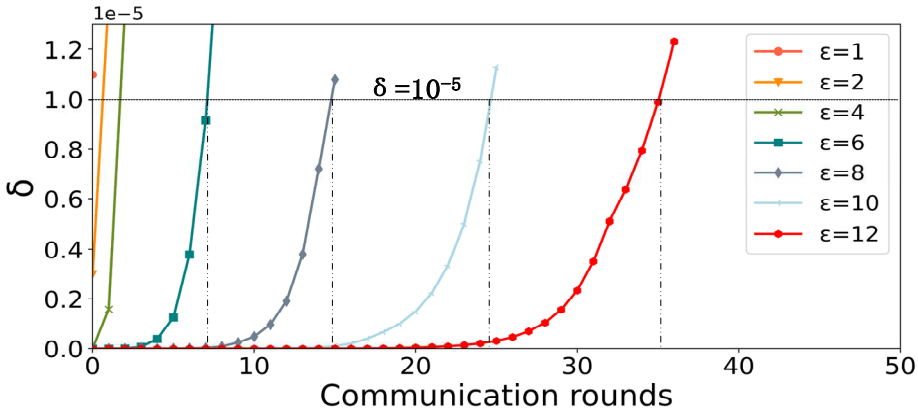
continue the communication rounds until δ reaches the threshold Q, then the whole training process will be stopped, and the server sends the well-trained model to all clients. As defined before, Q is selected as 1×10^{-5}. In this case study, the influence of different values of ε (7 values are chosen ranging from 1 to 12) on the model performance is investigated.

From Figure 5-7 (b), the DPFL-ATT-BLSTM scheme with small ε (1, 2, 4) reaches the threshold Q quickly within just a few communication rounds. However, the model accuracy is undesirable as nRMSE maintains a higher level than the Localized scheme, the benchmark model with the worst performance. At this privacy level, although consumers' privacy is protected perfectly, the functionality is ultimately sacrificed. In contrast, when ε is large enough (such as 10 or 12 in this case), it takes more communication rounds until σ reaches the threshold. More communication rounds allow the central model gets fully trained with frequent updates of its model parameters.

Consequently, the model accuracy increases as ε become larger. However, large ε allows less similarity of the outputs from different clients, and the adversary can distinguish different clients more effortless, and the model provides less privacy consequently. Hence, from the model accuracy results shown in Figure 5-7 (a), when ε between 6 and 8, the proposed scheme can efficiently make accurate load forecasting and provide a good level of privacy protection at the same time.



(a)



(b)

Figure 5-7. (a) model performance of the differential private federated learning scheme with different levels of privacy budget; (b) accumulation of total δ with increasing communication rounds under different privacy budgets.

5.4.10 Privacy and data ethics analysis

The privacy performance of the proposed scheme is discussed in the following aspects:

- *The proposed method satisfies* (ε, δ) *-DP.* In each communication round, the central server aggregates the client models by adding random noise to hide clients' contributions during training. The overall privacy cost is accumulated with the moment accountant method [364] given certain ε, σ and m. Once δ approaches the threshold Q (a special client's contribution to the output is too high) , the training process will stop immediately. Hence, it is proved that the system satisfies $(\varepsilon, \delta) - DP$.
- *The proposed method guarantees data privacy/confidentiality.* Referring to the adversary model defined in Chapter 4, TP is the honest-but-curious adversary in the smart metering system. Traditional value-added service platform requires the consumer to upload detailed high-resolution profiling data to the central server operated by a third party, which is the worst privacy invasion case. In contrast, the proposed model is constructed based on the FL framework, the individual power consumption data measured by the smart meter, which contains sensitive

information that never leaves the consumer's house. Hence, the service framework enhances data privacy/confidentiality for individual data items and strictly follows the data minimalization principle stated in the GDPR [301]. Moreover, the framework makes a trade-off between services and privacy/confidentiality, allowing consumers to select desirable TPSs without privacy concerns.

5.5 Chapter Summary

In this chapter, a novel privacy-preserving value-added service platform by considering both privacy and functionality requirements is proposed and validated. The platform is constructed based on the DPFL framework and utilizes a state-of-the-art ATT-BLSTM algorithm to train the local models. Moreover, the proposed system is based on the concept of global (ε, δ)-DP to balance the trade-off between privacy loss and model performance. In the case study of household-level STLF, the proposed scheme is evaluated with six benchmark models. After simulation, it is validated that the proposed system achieves prediction accuracy with low computation cost, and the privacy loss can be controlled flexibility by adjusting the privacy budget.

Part III Functionality

Chapter 6 Day-Ahead Distribution-Level Spectral Load Forecasting with Aggregated Smart Meter Data

6.1 Introduction

6.1.1 Motivation

The distribution-level Short-Term Load Forecasting (STLF) is a fundamental and important functionality for distribution-level grid management and monitoring. A reliable STLF is critical input information for Demand-Side Management (DSM), state estimation, maintenance scheduling, voltage support, etc. [365]. Moreover, providing precise and rapid prediction of future demands is the foundation of hourly-based applications such as electricity market-clearing mechanisms and regulation bids [366]. However, unlike the HV transmission network, the STLF for the distribution network is more challenging due to the high uncertainty of the low-capacity load, the diversity of users' characteristics, and the deep penetration of renewable energy [315]. Compared to the well-developed load forecasting at the HV network level, the distribution-level load forecasting method is still the technique at the exploratory stage. The aggregated smart meter data generated in the proposed smart metering system provides good data resources at the demand side to better forecast the demands at distribution level.

Conventional load forecasting techniques, such as Linear Regression (LR) and Auto-Regressive Integrated Moving Average (ARIMA), try to extract features in the time domain. Since various load components with different frequencies are contained in the load curve, the load demand is highly nonlinear and non-stationary. These characteristics of the original load demand make the prediction accuracy of

conventional models less accurate. In addition, AI-based load forecasting methods, especially RNN, have achieved desirable accuracy in recent years [159, 162]. RNN models have memory units that can learn current input features and information from the past. This characteristic is highly suitable for forecasting tasks. Although RNN can map nonlinear features like conventional approaches, it cannot learn frequency-domain information. Hence, a hybrid STLF method that can extract both time-domain and frequency-domain features with high adaptivity should be proposed.

6.1.2 Knowledge gaps

Although, as illustrated, there is a wealth of work available in the literature, the existing STLF models still have some knowledge gaps that can be filled.

- Although STLF has been fully investigated in transmission networks and household-level, distribution-level STLF is a relatively weak segment in current power systems.
- Electric spikes and other noises would influence the training process and the prediction accuracy, so a proper denoising technique should be selected to process the original data.
- The hybrid deep learning with Variational Mode Decomposition (VMD) and Empirical Mode Decomposition (EMD) in the literature either lacks mathematical definition or low adaptivity, respectively, so a new hybrid STLF takes advantage of both EMD and VMD should be proposed.

6.1.3 Contribution

This chapter proposes a novel hybrid distribution-level Denoising (DN)-EWT-BLSTM-Bayesian Hyperparameter Optimization (BHO) STLF algorithm, which combines mode decomposition with BLSTM to better extract the time-domain and frequency-domain features of the electric load. The contributions of this work are detailed as follows.

1) A hybrid STLF model that combines the EWT component decomposition with a BLSTM deep neural network is proposed to make multi-step predictions.

2) A wavelet-based denoising technique is proposed to eliminate the electric spikes.

3) A BHO method is proposed to find optimal hyperparameters with fast speed and adjust hyperparameters to different sub-layers.

6.1.4 Organization of the chapter

The remaining chapter is organised as follows: The DN-EWT-LSTM-BHO load forecasting algorithm is demonstrated in Section 6.2. In Section 6.3, four case studies are implemented, which compare the proposed load forecasting algorithm and other methods and evaluate the parameters that achieve the best performance. The conclusion and final discussion are provided in the last section.

6.2 Proposed Load Forecasting Algorithm

This section introduces the overall prediction system and the corresponding methodologies. The distribution network level electricity load data is obtained from the physical/informatic aggregator introduced in Chapter 4.

6.2.1 Overall forecasting model

As presented in Figure 6-1, the proposed method is divided into five steps and described as follows.

Step A: the first step is data pre-processing and denoising. The original electric load is input to the STLF model, and data cleaning is applied to the original dataset to populate the missing features. Then, a max-min scaling function is applied to the original dataset to limit the range of data between 0 and 1. Finally, a Discrete Wavelet Transform (DWT) based denoising algorithm is applied to the data to remove the noise.

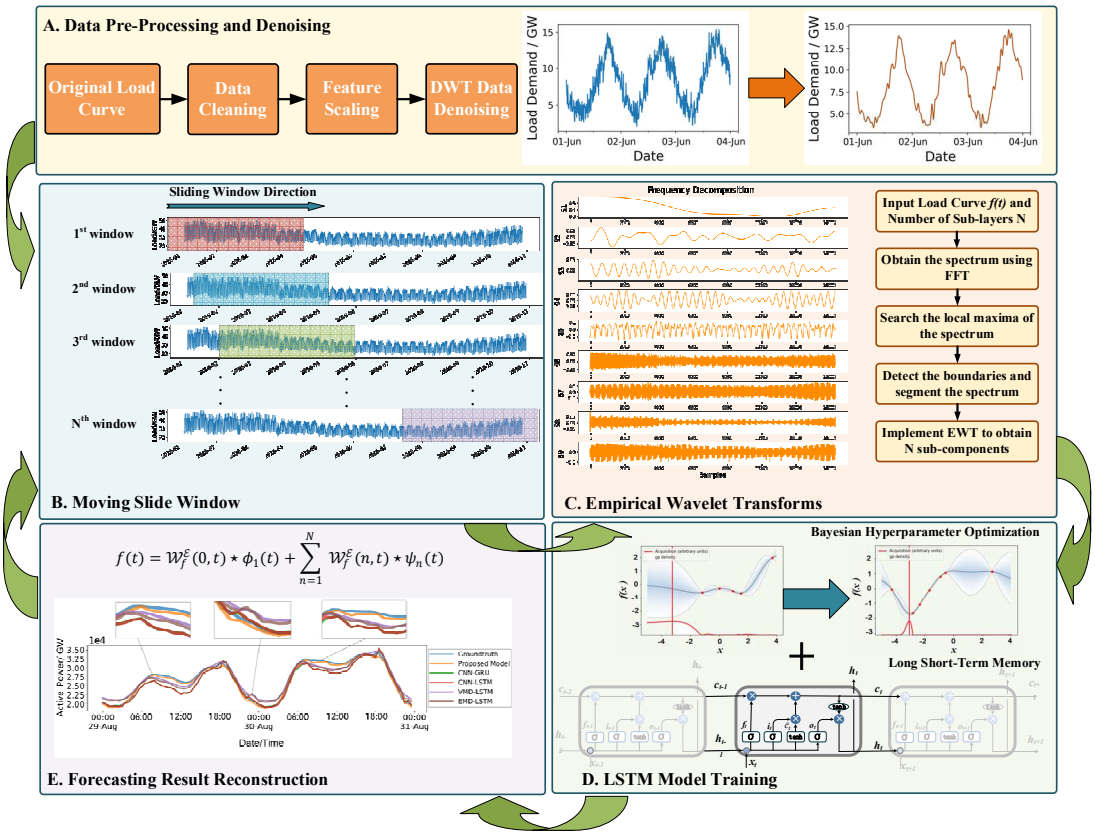


Figure 6-1. Overall process of the proposed spectral load forecasting model.

Step B: a sliding window is introduced to enable the proposed model to make real-time forecasts. The length of the sliding window is denoted as W, which is Wchosen as one week in this work. At the beginning of the training, the first W data is included in the window, and the model then predicts the load at $W + L_F$, where L_F is the forecasting step. Then, the sliding window will move smoothly and repeat the training process.

***Step C*:** the denoised electric load is decomposed into N sub-layers via the EWT decomposition algorithm, as indicated in Figure 6-1 (C); an example with nine sub-layers is presented to show the decomposed components from the original load curve.

Step D: then N BLSTM prediction models are constructed, and each BLSTM neural network model is trained for one sub-layer while the BHO method is employed to find the optimal hyperparameters.

***Step E*:** In the final step, the prediction results for all sub-layers are reconstructed to present the final load forecasting results. Repeat Steps A-D until reaching the end of the testing dataset.

6.2.2 Data description

The dataset employed in this chapter includes distribution-level electricity data, which is constructed by combining household-level smart meter data and weather and temporal data.

6.2.2.1 Distribution-level electricity data

In this chapter, the distribution network level data is obtained from the physical/informatic aggregator, and individual household-level smart meter data from Pecan Street Dataport (Dataport) [117] are added up to match the capacity of the feeder model. The geographical location of the elasticity data is *N* 30◦15'59.9976'', W 7◦43'59.9880'' (Austin, Texas, US), The feeder models used for this research are selected from standard feeder models provided by GridLAB-D [326], detailed description of the feeder models are introduced in Table 4-4 in Chapter 4. In this work, a total number of 976 houses are aggregated to match the R5-12.47-2 feeder model, indicating a moderate suburban area (demand capacity is 4500 kW). To reach the defined aggregation size, household smart meter data is picked up randomly from the Dataport dataset, an example of the demand load is shown in Figure 6-2.

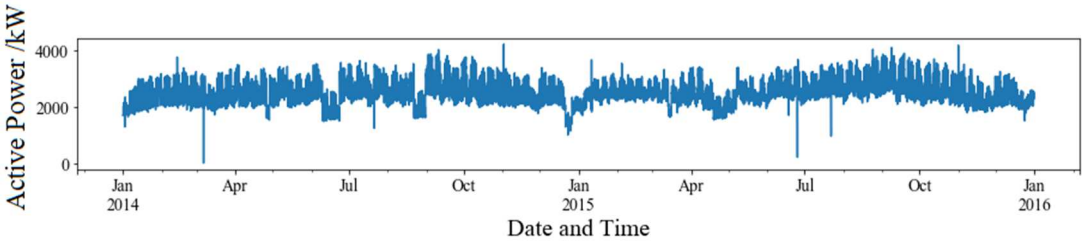


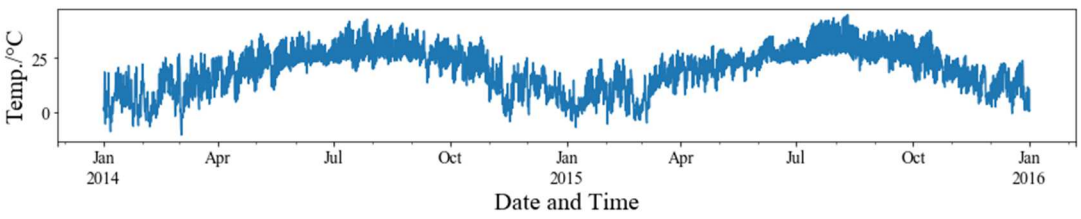
Figure 6-2. Active power of the distribution-level electricity data.

6.2.2.2 Weather and temporal information

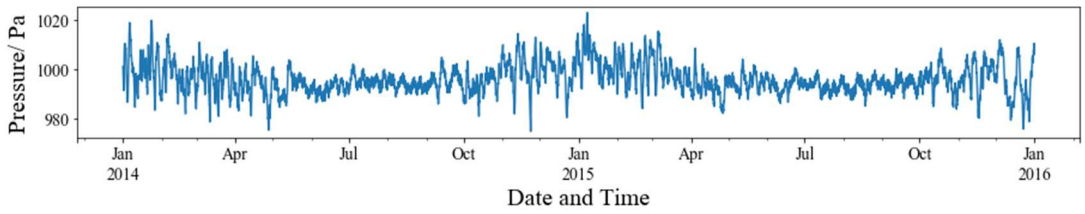
The corresponding weather and temporal information at the same location (N 30◦15’59.9976’’, W 7◦43’59.9880) are obtained from the National Solar Radiation Database (NSRDB) [195]. An example of the dataset is shown in Table 6-1 and Figure 6-3, and weather parameters include Dew point (℃), Temperature (℃), Pressure (Pa), and Relative Humidity (%RH). As for temporal information, four variables are introduced which are: Holiday (1 for holiday days and 0 for non-holiday days), Hour of the Day (HOD) (index range from 0 to 23), Day of the Week (DOW) (index range from 0 to 6), and Month of the Year (MOY) (index ranges from 1 to 12). As categorical variables, DOY, HOD, DOW, and MOY should be pre-processed by one-hot encoding.

Table 6-1 shows the weather and temporal dataset [195].

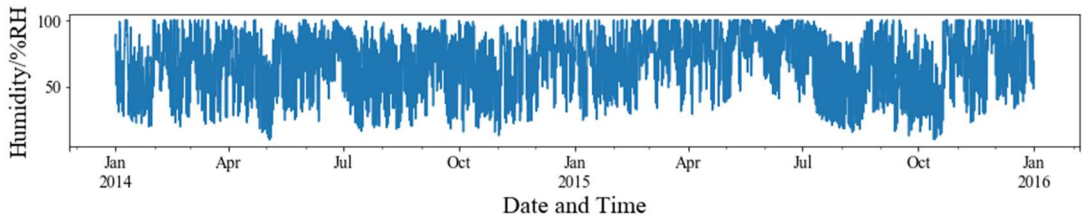
Timestamp	Holiday	HOD	DOW	MOY	Dew. Point (℃)	Temperature (℃)	Pressure (Pa)	Relative Humidity (%RH)
01/01/2014 00:00	1	0	3	1	-1.25931	1.801934814	1001.035	80.1306
01/01/2014 01:00	1	1	3	1	-1.25199	1.385064697	1000.496	82.60188
01/01/2014 02:00	1	2	3	1	-1.25886	1.022241211	999.9987	84.73964
01/01/2014 03:00	1	3	3	1	-1.26002	0.723382568	999.3622	86.57533
01/01/2014 04:00	1	4	3	1	-1.23658	0.513696289	998.8751	88.04627
01/01/2014 05:00	1	5	3	1	-1.19007	0.407220459	998.4365	89.02894
01/01/2014 06:00	1	6	3	1	-1.09016	0.479547119	998.138	89.215
01/01/2014 07:00	1	7	3	1	-0.80314	1.532342529	998.006	84.46244



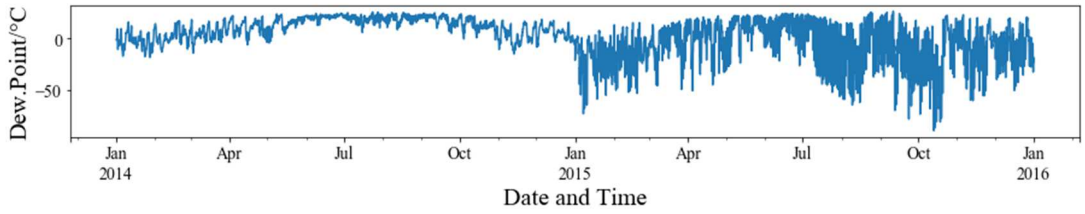
(a) Temperature.



(b) Pressure.



(c) Humidity.



(d) Dew Point.

Figure 6-3. Visualization of the Weather variables.

6.2.3 Data denoising with wavelets

The original load data contain a significant amount of noise generated from various sources, such as the electric spikes of electric appliances and intermittent penetration of distributed generators. In addition, the measurement devices such as smart meters, DSCADA, and μPMU also produce electronic noise. The high-frequency noise in the measured feeder load demand is a severe issue that influences the performance of load prediction. DWT could effectively analyze the non-stationary signals and reduce the high-frequency noise [367].

The theory of the DWT-based denoising technical is to decompose the original data into the high-frequency and low-frequency components, and a suitable threshold of the high-frequency components is determined for denoising purposes. Finally, the signal is reconstructed again. Sampling the original data $f^*(t)$ with frequency f_{sample}

to obtain the discrete signal $f^*(m)$, $m = 1,2 \ldots, M$. The purpose of the signal denoising is to remove noise and find the best estimation of the underlying signal $f(m)$:

$$f^*(m) = f(m) + \sigma_{noise}\epsilon_m, m = 1,2, \ldots, M \tag{6-1}$$

where ϵ_m is the white Gaussian noise; and σ_{noise} shows the noise intensity, and M is the total sample number of the discrete form of the signal. A two-level DWT wavelet decomposition process is shown in Figure 6-4. From the figure, the signal can be decomposed into two coefficients: approximation coefficients (a) and the detailed coefficients (d). At the first level, $f^*(m)$ is decomposed into a_1 and d_1, and a_1 is then decomposed into a_2 and d_2 further.

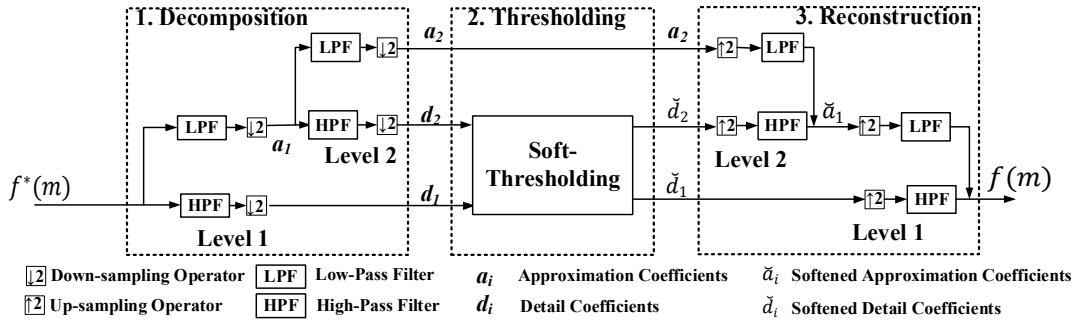


Figure 6-4. Block diagram of signal denoising with wavelets.

The denoising approach includes three steps: signal decomposition, denoising, and reconstruction. The decomposition level $S = 2$ in this thesis.

***Step 1*: decomposition**. The original load demand, the noisy signal, is decomposed via the DWT, as shown in Figure 6-4. The original signal is passed through a series of High-Pass Filters (HPFs) and Low-Pass Filters (LPFs). The i th level detailed coefficients $d_i(k)$ are given via an HPF and the ith level approximation coefficients $a_i(k)$ are given via an LPF. The decomposition functions of the ith level are the following:

$$a_i(k) = \sum_m f^*(m)\varphi_g(2m - k) \tag{6-2}$$

$$d_i(k) = \sum_m f^*(m)\varphi_h(2m - k) \tag{6-3}$$

where φ_h and φ_g are the functions of HPF and LPF, respectively, and k is the translation factor.

***Step 2*: denoising**. It is essential to determine a suitable threshold *Thr* for data denoising, and a thresholding function $\rho_T(x)$ is required. Thresholding can be divided into hard and soft thresholding. For hard thresholding, the values that exceed the threshold would be set to 0. The magnitude of coefficients greater than the threshold for soft thresholding is softened. The noise level δ_{mad} is first estimated from the detail coefficients by median absolute deviation, as follows:

$$\delta_{\text{mad}} = \frac{\text{median}\{d_i\}}{0.6745} \tag{6-4}$$

$$Thr = \delta_{\text{mad}}\sqrt{ln(M)} \tag{6-5}$$

After the threshold *Thr* is determined, the soft thresholding function is applied to reduce the magnitude of the coefficient, which is defined as:

$$\rho_T(x) = \begin{cases} x - Thr & \text{if } x \geq Thr \\ x + Thr & \text{if } x \leq -Thr \\ 0 & \text{if } |x| \leq Thr \end{cases} \tag{6-6}$$

***Step 3*: reconstruction.** The coefficients after the soft thresholding are reconstructed via Inverse Discrete Wavelet Transform (IDWT). In Figure 6-1, *Step 1* compares the original load demand curve and the denoised demand curve. It is observed that the noise and spikes from the original data are successfully cleared.

6.2.4 Empirical Wavelet Transforms (EWT)

After the data is denoised via DWT, the denoised data $f(t)$ is decomposed into N sub-layers via EWT. In [368], EWT aims to extract multiple sub-layers by constructing adaptive wavelets. The EWT decomposition process is performed in the following steps. In EWT decomposition, the number of sub-layers N is defined at the beginning.

Step 1: apply Fast Fourier Transform (FFT) to the denoised data $f(t)$ to obtain the frequency spectrum $F(\omega)$.

Step 2: search the $F(\omega)$ to find N local maxima $\boldsymbol{\partial} = \{\partial_n\}_{n=1,2\ldots,N}$ and corresponding frequencies $\boldsymbol{\omega} = \{\omega_n\}_{n=1,2\ldots,N}$ by using the magnitude threshold α and frequency distance thresholds δ. α is set as 3% of the fundamental magnitude to detect the significant frequencies, and δ is set as 8 Hz to avoid overestimation [369].

Step 3: segment the frequency spectrum $[0, f_{sample}/2]$ into N segments, and the boundaries Ω_n is the centra line between two neighbouring local maxima (see Figure 6-5), which can be calculated as:

$$\Omega_n = \frac{\omega_n + \omega_{n+1}}{2} \tag{6-7}$$

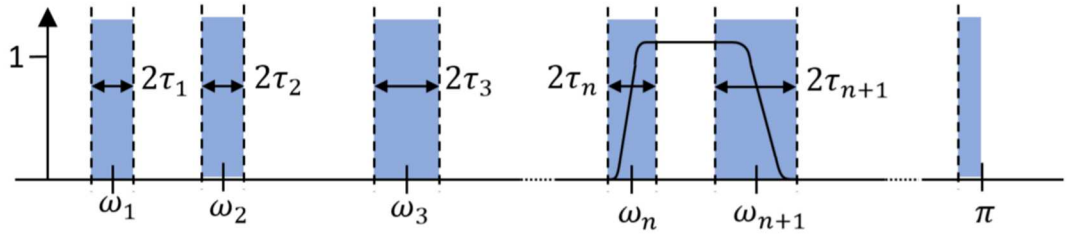


Figure 6-5. Segmenting Fourier spectrum into N contiguous segments (Adopted from [370]).

Step 4: build N wavelet filters, including one low-pass filter and $N-1$ band-pass filters based on the defined boundaries. The scaling and wavelet functions are defined in (6-8) and (6-9), respectively.

$$\hat{\phi}_n(\omega) = \begin{cases} 1, & \text{if} |\omega| \le (1-\gamma)\omega_n \\ cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}(|\omega| - (1-\gamma)\omega_n)\right)\right], & \\ \text{if}(1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n & \\ 0, \text{otherwise} & \end{cases} \tag{6-8}$$

$$\hat{\psi}_n(\omega) = \begin{cases} 1,\text{if}(1+\gamma)\omega_n \le |\omega| \le (1-\gamma)\omega_{n+1} \\ cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_{n+1}}(|\omega| - (1-\gamma)\omega_{n+1})\right)\right], \\ \text{if}(1-\gamma)\omega_{n+1} \le |\omega| \le (1+\gamma)\omega_{n+1} \\ sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}(|\omega| - (1-\gamma)\omega_n)\right)\right], \\ \text{if}(1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n \\ 0,\text{otherwise} \end{cases} \tag{6-9}$$

where the arbitrary function β and the ratio γ are defined as :

$$\beta(x) = \begin{cases} 0 & \text{if } x \le 0 \\ & \text{and } \beta(x) + \beta(1-x) = 1 \forall x \in [0,1] \\ 1 & \text{if } x \ge 1 \end{cases} \tag{6-10}$$

$$\gamma < \min_n \left(\frac{\omega_{n+1}-\omega_n}{\omega_{n+1}+\omega_n}\right) \tag{6-11}$$

Step 5: perform scaling and wavelet functions shown in (6-12) and (6-13) to extract the approximate and detailed coefficients.

$$\mathcal{W}_f^{\varepsilon}(0,t) = \langle f, \phi_1 \rangle = \int f(\tau)\overline{\phi_1(\tau - t)}d\tau = \left(\hat{f}(\omega)\overline{\hat{\phi}_1(\omega)}\right)^{\vee} \tag{6-12}$$

$$\mathcal{W}_f^{\varepsilon}(n,t) = \langle f, \psi_n \rangle = \int f(\tau)\overline{\psi_n(\tau - t)}d\tau = \left(\hat{f}(\omega)\overline{\hat{\psi}_n(\omega)}\right)^{\vee} \tag{6-13}$$

where $\langle\,\rangle$ stands for the inner product $\wedge$ and $\vee$ indicates the Fourier transform and its inverse, $\overline{\phi_1(\tau - t)}$ and $\overline{\psi_n(\tau - t)}$ are the conjugate complex numbers of $\phi_1(\tau - t)$ and $\psi_n(\tau - t)$, respectively.

Step 6: compute the sub-band signals. The approximation sub-band signal $f_0(t)$ and the n^{th} detail sub-band signal $f_n(t)$ can be computed by (6-14) and (6-15).

$$f_0(t) = \mathcal{W}_f^{\varepsilon}(0,t) \star \phi_1(t) \tag{6-14}$$

$$f_n(t) = \mathcal{W}_f^{\varepsilon}(n,t) \star \psi_n(t) \tag{6-15}$$

where $\star$ denotes the convolution operation.

The EWT reconstruction, also called Inverse Empirical Wavelet Transform (IEWT), is used to reconstruct the sub-layers to $f(t)$. $f(t)$ can be reconstructed via the reconstruction function as follows:

$$\begin{aligned} f(t) &= f_0(t) + \sum_{n=1}^{N} f_n(t) \\ &= \mathcal{W}_f^{\varepsilon}(0,t) \star \phi_1(t) + \sum_{n=1}^{N} \mathcal{W}_f^{\varepsilon}(n,t) \star \psi_n(t) \\ &= \left(\widehat{\mathcal{W}_f^{\varepsilon}}(0,\omega)\hat{\phi}_1(\omega) + \sum_{n=1}^{N} \widehat{\mathcal{W}_f^{\varepsilon}}(n,\omega)\hat{\psi}_n(\omega)\right)^{\vee} \end{aligned} \tag{6-16}$$

6.2.5 Bayesian hyperparameter optimization

Training and optimizing a deep learning model are complex process that involves a great number of hyperparameters and regularization terms. Hyperparameter optimization is essential for training neural networks as it aims to find the hyperparameters that return the best accuracy or performance given a dataset. However, the hyperparameter tuning process is normally a 'black box' function, which requires the examiners to keep querying the model and obtain feedback on model performance. The hyperparameter optimisation problem for a 'lack box' function $G(x)$ can be formalized as:

$$x_M = \arg \min_{x \in X} G(x) \tag{6-17}$$

where x_M is the optimal hyperparameter set, and X is the candidate set. The target of the function is to find x_M which can minimize $G(x)$. Grid search is the most fundamental hyperparameter tuning method [371], where space is defined for each hyperparameter at first, and then the algorithm exhaustively searches this space sequentially and trains a model for every possible combination of hyperparameter values. The drawback of the grid search method is that the number of training models increase exponentially when hyperparameters increase.

Compared to the above methods, a novel BHO was proposed in 2011 [372]. Instead of searching the hyperparameter space blindly, BHO creates a prior distribution model, and then the model is optimized with the given information to fit the actual

distribution better. Furthermore, it can use the results from the previous iteration to decide the next candidate value of the hyperparameter. Hence, the BHO is much more efficient and less time-consuming as it selects the optimal hyperparameter in an informed manner and better utilizes the past information.

The central methodology of the BHO method is to construct a surrogate probability model to select hyperparameters to minimize the original objective function. Providing a sample domain $\mathcal{X}$, the true objective function $G(x)$ to be optimized is approximated with a surrogate function $\mathcal{M}$. $\mathcal{M}$ is initialized with a small data group from $\mathcal{X}$, and an acquisition function $\mathcal{S}$ is adopted to choose the next point to query. A variety of surrogate functions $\mathcal{M}$ is introduced in [373], including Gaussian Processes (GPs), random forests, and Tree-Structured Parzen Estimators (TPEs) [33]. In this work, GP is employed as the surrogate function. The GP is a stochastic process that collects random variables in the time or space domain, such that each linear finite-dimensional restriction is a joint Gaussian distribution [374]. A GP is restricted by a mean $\mu(x)$ and a covariance function $k(x, x')$, while $\mu(x)$ is assumed to be zero in most situations, and $k(x, x')$ determines the smoothness of $G(x)$. $k(x, x')$ is regarded as the kernel of GP and needs to be symmetric, continuous, and positive, and the square exponential function is employed as the kernel in most cases:

$$k(x, x') = l \cdot exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \tag{6-18}$$

where l and σ are the positive parameters.

As for $\mathcal{S}$, it determines the next point to query by selecting the most promising candidate. Normally, three acquisition functions are widely used, which are the Maximum Probability of Improvement (MPI) [375], Expected Improvement (EI) [376], and Upper Confidence Bound (UCB) [377]. The disadvantage of MPI is that it only chooses the points with high confidence to query. Hence there is little improvement in the model. EI overcomes the limitation of MPI by maximizing the expected improvement over the current best. In such a way, if the new value performs much better, the model improves a lot; if the new value performs much worse, the

model maintains the same. In this work, EI is chosen as $\mathcal{X}$. The formula of EI is expressed as:

$$acqu_{EI}(x;\{x_n,y_n\},\theta)=\sigma(x;\{x_n,y_n\},\theta)\left(\gamma(x)\Phi\big(\gamma(x)\big)\right)++N(\gamma(x);0,1) \quad (6\text{-}19)$$

$$\gamma(x)=\frac{f(x_{\text{best}})-\mu(x;\{x_n,y_n\},\theta)}{\sigma(x;\{x_n,y_n\},\theta)} \quad (6\text{-}20)$$

where x_{best} is the best value at the current stage; θ is the parameters of the GP model; x_n and y_n are the available samples; $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal; $\mu(x;x_n,y_n,\theta)$ denotes the predictive mean function; $\sigma(x;x_n,y_n,\theta)$ denotes the predictive variance function. The detailed BHO process is illustrated in Algorithm 6-1.

Figure 6-6 and Algorithm 6-1 illustrate the Bayesian optimization procedure over three iteration; the black dash line indicates the objective function, which is unknown to the examiners. The upper blue shaded curve is the confidence interval generated by a probabilistic estimation model of the objective function. Furthermore, the lower green shaded curve represents the acquisition function. A high value of the acquisition function means high prediction uncertainty (exploration) and high objective (exploitation) [378]. The location of the peak point of the acquisition function will be selected as the new point to query in the objective function. With the increasing observation points, the posterior distribution improves, and the posterior uncertainty decreases.

Algorithm 6-1: Bayesian hyperparameter optimization with Gaussian Process.

1: **For** t=1,2,…
2: Find the new x_{t+1} by maximizing acquisition function $\mathcal{S}$:
3: $$x_{t+1}=arg\max_{x}\mathcal{S}(x|\mathcal{D}_t)$$
4: Query the objective function to obtain $y_{t+1}=f(x_t)+\varepsilon_t$.
5: Argument the data $\mathcal{D}_{t+1}=\{\mathcal{D}_t,(x_t,y_t)\}$.
6: Update the Gaussian Process model.
7: **End for**.

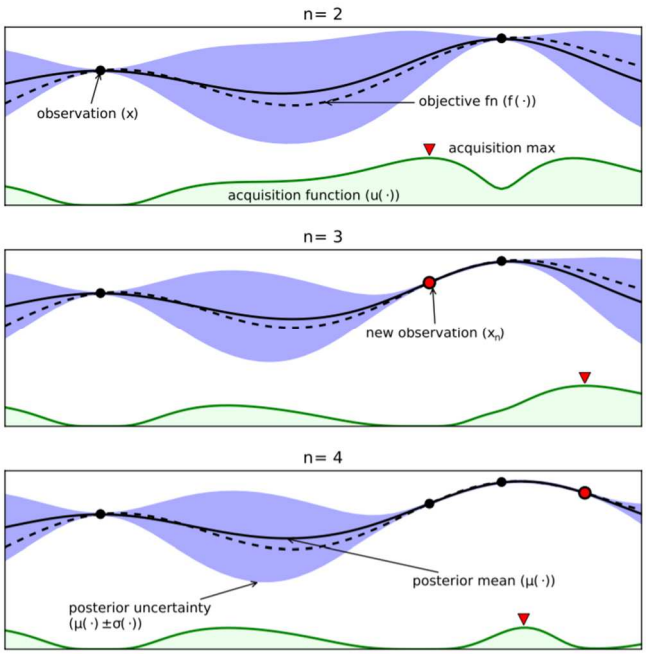


Figure 6-6. Illustration of the Bayesian optimization procedure over three iterations (Adopted from [378]).

6.3 Experimental Setup

6.3.1 Open access software platform and package

Various open access packages and libraries based on Python 3.7 and TensorFlow 2 are adopted to implement the proposed simulation case study. PyWavelets [379], PyEMD [380], ewtpy [381], and vmdpy [381] are used for implementing DWT, EMD, EWT, and VMD, respectively. A Bayesian hyperparameter optimization package, Hyperopt [382], is used for hyperparameter tuning.

6.3.2 Performance metrics

To assess the performance of the proposed predictor, the following four performance metrics are adopted: MAE, MAPE, RMSE, and R^2. The detailed formulas are introduced in Equations (3-14) - (3-17) in Chapter 3.

6.4 Results and Discussion

To evaluate the proposed load forecasting model, the ND-dataset, as mentioned earlier, and EWD-dataset are tested. Three case studies are designed in this section, i.e., the influence of several sub-layers N on the forecasting performance, the computation time of BHO and relevant hyperparameter tuning methods, and a comparison between the proposed model and relevant works.

6.4.1 Case study I: Impact of the number of sublayers N

Referring to the EWT decomposition technique introduced in Section 6.2, the original time-varying load demand is decomposed into N sub-layers by the EWT, which is defined as $S_1 - S_N$ in this study. The number of N has a significant impact on the final forecasting performance. In this study, the range of N increases from 5 to 13. Both the ND-dataset and the EWD-dataset are used in the comparison experiment. The performance of the proposed model with different numbers of N is summarized in Table 6-2 and Figure 6-7. From these tables, it is observed that the MAE, MAPE, and RMSE are relatively large when N is too tiny (near 5) or too large (near 13) (see Figure 6-7). Among all N values, the dominant value is $N = 10$, followed by $N = 9$, where the RMSE values are 101.089 kW and 102.900 kW, respectively.

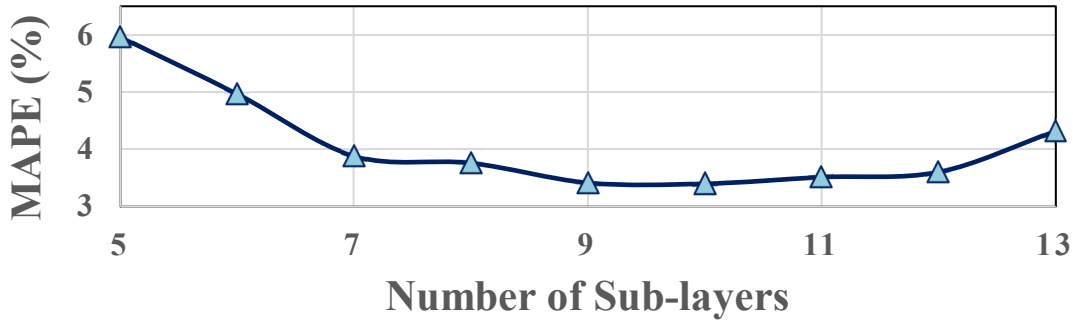


Figure 6-7. MAPEs of the proposed model with different sub-layer numbers.

Table 6-2. Day-ahead prediction performance of the proposed model with different sublayer numbers.

N	MAE (kW)	MAPE (%)	RMSE (kW)	R^2
5	146.721	5.959	202.909	0.725
6	121.614	4.965	159.892	0.837
7	93.622	3.878	124.382	0.928
8	90.313	3.762	116.665	0.936
9	80.222	3.416	102.900	0.954
10	**79.948**	**3.398**	**101.089**	**0.956**
11	83.153	3.517	106.233	0.947
12	84.610	3.604	105.636	0.949
13	100.364	4.318	122.169	0.931

Once the optimal number of the decomposition layers is determined, the denoised load demand data is decomposed by EWT to obtain the sub-components. Then, N LSTM predictors are trained simultaneously to predict each sub-component. The predictions for decomposed sublayers given the validation set are shown in Figure 6-8. The load demand is decomposed into N sub-layers by the EWT, which provides the best performance of the selected datasets ($N = 10$). Sub-layers ($S_1 - S_3$) capture the low-frequency oscillation of the baseline, and the curves of these sub-layers vary smoothly and change steadily. The predicted curves of these four layers achieve higher accuracy from the prediction results. While the Sub-layers ($S_8 - S_{10}$) capture high-frequency components which have a high fluctuation range and include most noise, and most prediction errors come from the prediction for these components.

6.4.2 Case study II: Impact of weather information

In this case study, the model with weather information input is compared with the model without weather information. As introduced in Section 6.2.2, relevant weather variables are employed as the input variables of the proposed STLF model. A comparison is made among STLF without external information, STLF with weather information, and STLF with both weather and temporal information, see Table 6-3. From the table, it is observed that the model without external information achieves the lowest prediction accuracy. When the weather information, e.g., temperature, humidity, and the dew point, is added as input variables, the prediction accuracy improves RMSE by 6.34%, MAE by 7.94%, and MAPE by 7.89%, respectively. Whilst the introduction of the temporal information, e.g., DOY, HOD, DOW, and MOY, the prediction performance improves further, which demonstrates that the relevant variables can enhance the prediction accuracy.

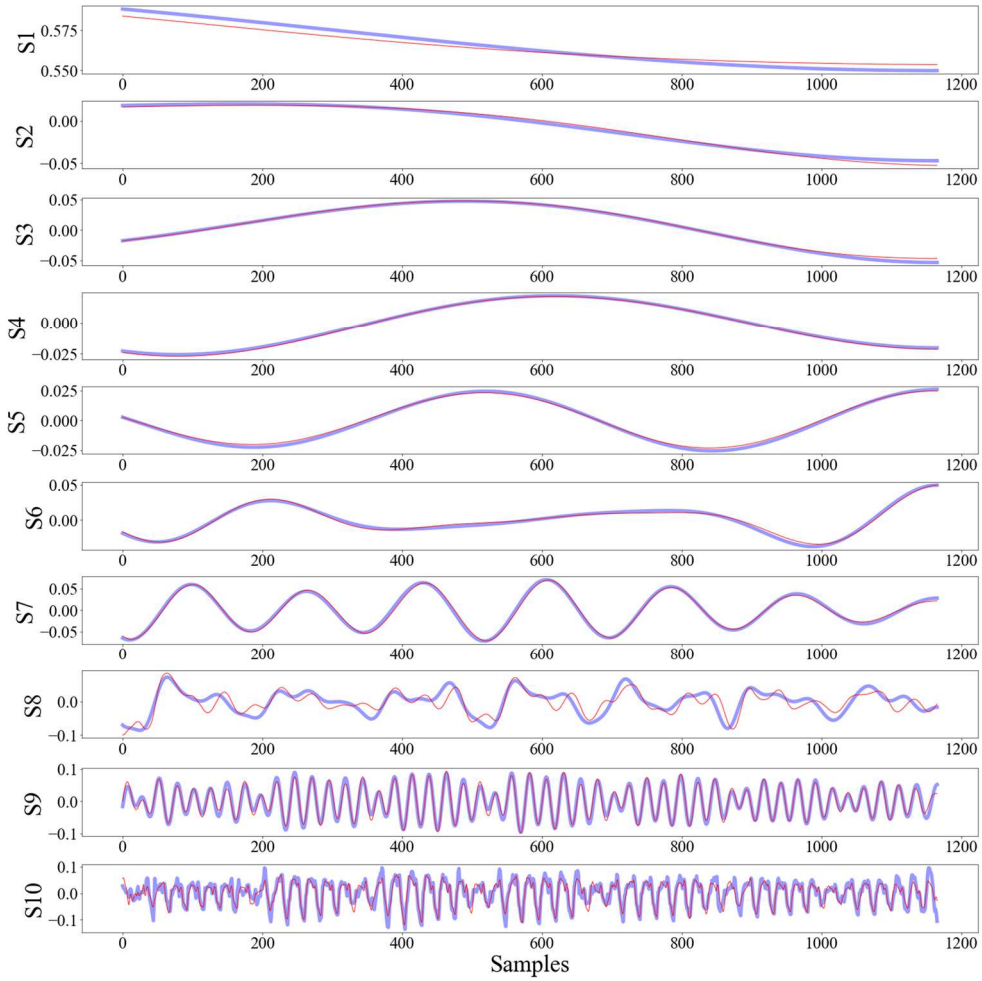


Figure 6-8. Validation for each sublayer in the validation set.

Table 6-3. Comparison of methods with/without weather information.

Method	MAE (kW)	MAPE (%)	RMSE (kW)	R2
Model + Weather Information	73.602	3.130	94.676	0.962
Model + Weather Information + Temporal Information	**70.666**	**3.004**	**91.084**	**0.966**
Model without External Information	79.948	3.398	101.089	0.956

6.4.3 Case study III: Comparison of BHO with grid searching and the random search

In this case study, the proposed BHO method is compared with two hyperparameter tuning approaches, i.e., grid search and random search. As the naive hyperparameter tuning approach, grid search simply searches the whole hyperparameter space, defined

in Table 6-4. Another tuning method, i.e., random search [47], tunes the hyperparameters by randomly selecting the combinations of possible parameters. As the proposed hybrid STLF method trains N BLSTM sub-models in parallel, the optimal hyperparameters should be evaluated for all sub-models. Moreover, the sub-layer number is selected as the optimal value evaluated in Case study I. This work's hyperparameters include learning rate, dropout rate, cell type, number of hidden layers, epochs, etc.

Table 6-4. Hyperparameter tuning range.

Hyperparameter	Range
Learning rate	$10^{-5} \sim 10^{-1}$
Dropout rate	0.3 ~ 0.7
Cell type	GRU, LSTM
Number of hidden layers	1 ~ 5
Batch size	32, 64, 128, 256, 512, 1024
Optimizer	Adam, Nadam, RMSprop, Adagrad
Loss	MSE, MAPE, MAE, Huber
Activation function	ReLU, Sigmoid, Tanh
Epochs	20, 50, 100, 150, 200

Both the prediction accuracy and training time are compared in Table 6-5. From the tables, it is found that although the traditional grid search method achieves almost equal prediction accuracy as BHO, it is time-consuming; the disadvantage of the grid search method would be more obvious when the hyperparameter space is large or the structure of the neural network is complex. As for the random search method, it costs the shortest time, but the prediction accuracy decreases as well. The proposed BHO method takes advantage of both grid search and random search, giving second-optimal results with a much faster computation speed than the grid search method.

Table 6-5. Results of different hyperparameter optimization methods.

Method	MAE (kW)	MAPE (%)	RMSE (kW)	R2	Time
Grid Search	74.563	3.245	95.213	0.962	63h32min
Random Search	85.672	3.641	104.726	0.957	12h24min
BHO	70.666	3.004	91.084	0.966	13h12min

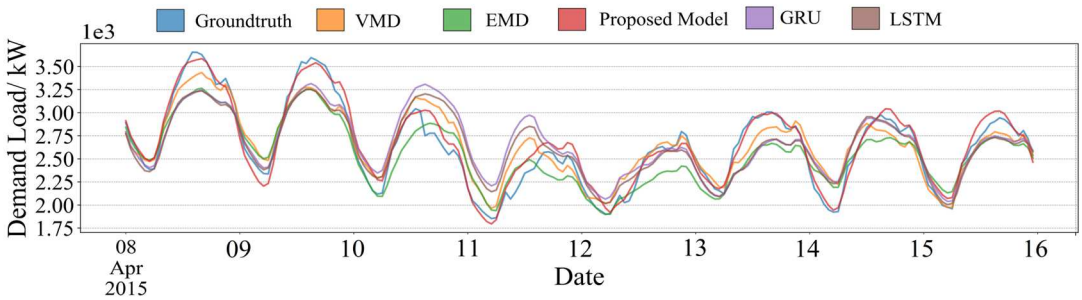
6.4.4 Case study IV: Comparison of the performance of the proposed method with other algorithms

In this case study, the one-step forecasting performance of the proposed method is compared with relevant forecasting approaches. A detailed description of the models adopted in this study is listed below: ① 1D CNN-LSTM STLF model; ② 1D CNN-

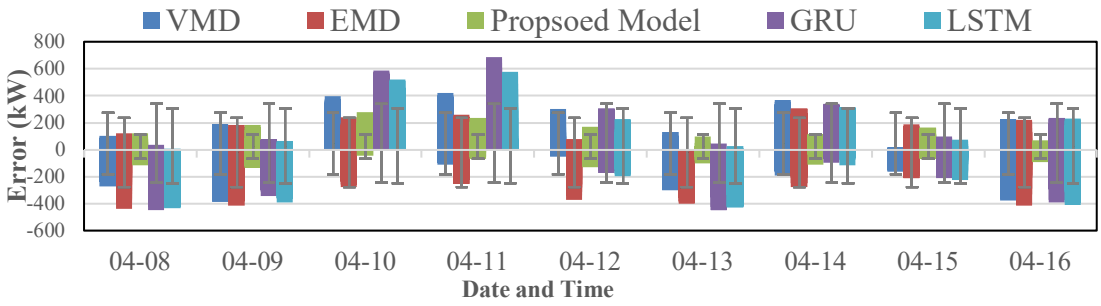
GRU STLF model; ③. EMD-LSTM STLF model; ④ VMD-LSTM STLF model; ⑤ ND-EWT-BLSTM-BHO STLF model (proposed model).

For models ① and ②, the original time-varying load demand is adopted as the input of neural network models. While for models ③, ④, and ⑤, the original load demand data are decomposed via EMD/VMD/EWT, respectively, and then the neural network is trained for each sub-layer.

Table 6-6 shows the performance of five models considering the performance metrics, i.e., MAE, MAPE, RMSE, and R^2, of the predicted load demand given the distribution-level dataset. As shown in the table, the proposed ND-EWT-LSTM-BHO outperforms other models. Moreover, the spectral load forecasting methods, including ND-EWT-LSTM-BHO, EMD-LSTM, and VMD-LSTM, have better prediction accuracy than conventional deep learning methods, including 1D CNN-LSTM and 1D CNN-GRU. 1D CNN-LSTM and 1D CNN-GRU models have the worst estimation performance with the highest MAE, MAPE, and RMSE in all experiment groups. The prediction performance of VMD-LSTM and EMD-LSTM are similar, just below the proposed method. Figure 6-9 compares the predicted values with the testing set using the proposed and benchmark models. The results predicted by the proposed model are the closest to the ground truth measurements. Moreover, the results estimated by the CNN-LSTM/CNN-GRU model are farthest from the ground truth curve, showing that CNN-LSTM and CNN-GRU perform worst among all algorithms.



(a) Load forecasting result.



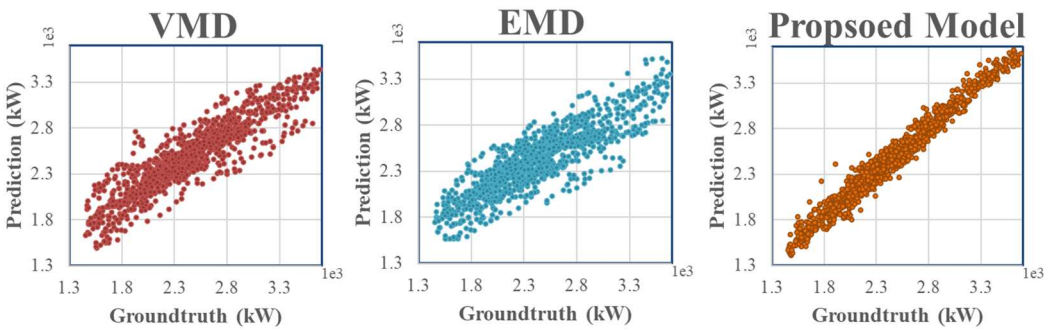
(b) Forecasting error.

Figure 6-9. Day-ahead forecasting results on distribution-level load. (a) Load demand profiles. (b) Load demand forecasting error.

Figure 6-10 shows the scatter plot of different forecasting models' ground truth and forecasting values. The scatter plot shows the correlation relationship between the two variables. The higher the R^2 value, the stronger the correlation between the predictions and ground truth, representing better accuracy achieved by the forecasting model. For the proposed model, the scatter about the line is relatively small, and most points are on the regression line, with only several data values far from other data values. For other spectral methods, the R^2 of VMD-LSTM and EMD-LSTM models also show a strong correlation with the ground truth curve, with R^2 values over 0.70. CNN-GRU shows the worst correlation from the scatter plot, with R^2 values of 0.429.

Table 6-6. Prediction performance of the proposed model and related works (ND-dataset).

Method	MAE (kW)	MAPE (%)	RMSE (kW)	R2
1D CNN-LSTM	189.822	8.564	267.284	0.487
1D CNN-GRU	205.014	9.270	284.339	0.429
VMD-LSTM	122.899	5.010	171.473	0.803
EMD-LSTM	150.303	6.286	196.932	0.709
Proposed Method	**70.666**	**3.004**	**91.084**	**0.966**



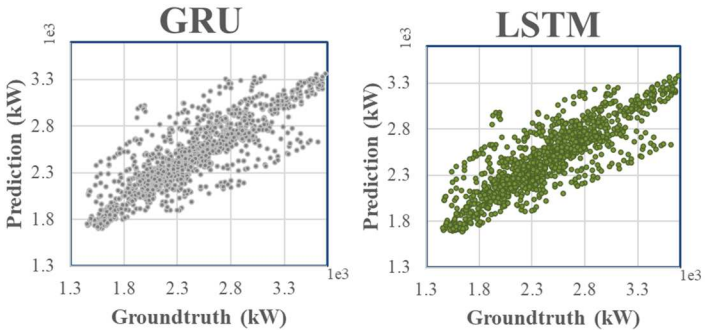


Figure 6-10. High-density scatter plot of ground truth and prediction values of day-ahead load forecasting models.

6.4.5 Discussion

In this subsection, four case studies are presented. The main findings are summarized as follows.

1) The first case study investigates the influence of the sub-layer number N on the prediction performance. Referring to the simulation, it is observed that when N equals 10, the proposed load forecasting model achieves the highest accuracy. When N is too small, the high-frequency components cannot be captured completely, and a large N will introduce the complexity of the forecasting model, which requires higher computation capacity and longer computation time to train the model.
2) Case study II demonstrates that the forecasting model with external weather and temporal information achieves higher prediction accuracy than the naive model. This is due to the consumer's electricity usage being highly related to these variables.
3) Case study III examines the BHO hyperparameter tuning approach by comparing it which traditional grid search and random search methods. The results show that the BHO approach takes the advantage of the high accuracy of the grid search method and the rapid speed of the random search method.
4) In the last case study, the optimized model is compared with other forecasting models, i.e., 1D CNN-LSTM, 1D CNN-GRU, VMD-LSTM, and EMD-LSTM.

The results show that the proposed model improves RMSE by 28.01%, MAE by 34.11%, and MAPE by 28.92% for the distribution-level dataset, respectively.

6.4.6 Contribution to privacy

Existing household-level and distribution-level STLF methods require close monitoring of the electricity usage from individual households via the smart meters, which violates privacy (as introduced in Chapter 4). The proposed distribution STLF method only uses the aggregated electricity data from the physical/informatic aggregator to make the prediction. As a result, the distribution network is well predicted without inferring an individual's personal information.

6.5 Chapter Summary

Accurate day-ahead load forecasting is extremely important for demand-side management and power planning. In this chapter, a hybrid load forecasting model ND-EWT-BLSTM-BHO is proposed by extracting both time-domain and frequency-domain information to reduce the uncertainty of load forecasting. The model considers the wavelet-based denoising algorithm, EWT component decomposition technique, BLSTM algorithm, and BHO algorithm. The proposed model first filters noise such as electric spikes from the measured load demand data. Then, an EWT algorithm is adopted to decompose the data into N sub-layers to extract time and frequency domain features. N LSTM neural network models are trained for all sub-layers as the next step.

Additionally, a BHO algorithm tunes the hyperparameters to find the best combinations that achieve the best performance. Finally, the prediction results for all sub-layers are reconstructed and present the result of the load forecasting. The distribution load demand data, aggregated from the household-level smart meter readings, are used for the simulation. In this chapter, four case studies are demonstrated. The conclusion is that the proposed model performs better than existing component decomposition models.

Chapter 7 A Feeder-Level Solar Energy Decoupling Scheme with Aggregated Smart Meter Data

7.1 Introduction

7.1.1 Motivation

Detecting the distributed renewable generation under the feeder/distribution network is another significant function required by the DNO. In recent years, the deep penetration of distributed renewable generation, especially rooftop solar energy, has brought new challenges for the DNO to monitor the distribution network. Although high PV penetration reduces greenhouse gas emissions and leads to an environmentally friendly world, it also significantly changes the existing power system structure. It is vital to increase the visibility of these renewable energy generations to manage the power system better. Among all installed solar panels, small-scale or rooftop PV occupies nearly 50% of the overall capacity. Unlike the large-scale PV stations measured individually, most rooftop PV is behind the meter, which means that the power generated by PV cannot be recorded by the smart residential meter. A lack of visibility would limit demand-side management, including scheduling the short-term operations implemented by grid operators. The conventional method requires installing an electricity meter beside the solar panel of each house, which requires extra measurement devices and communication channels. These devices largely increase budgets and invade personal privacy [383]. Moreover, the ownership of these devices also raises conflicts among stakeholders. With the aggregated smart meter data provided by the proposed smart metering system, it is

possible to construct the distribution network and the feeder model and decouple the overall PV generation under this network. The mathematical model of the problem to be solved in this chapter is introduced as follows.

7.1.2 Problem statement

The feeder system is shown in Figure 7-1. The feeder connects a few houses and unmonitored rooftop PV systems. The power utility has the authority to access the real-time grid measurement of a feeder/substation (active power $P_{Net}(t)$, reactive power $Q_{Net}(t)$ etc.). The components of the feeder include the total demand load of all residences served by the feeder, P_{Load}; and the power generated by the rooftop PVs in this area, P_{PV}:

$$P_{Net}(t) = P_{Load}(t) - P_{PV}(t) \tag{7-1}$$

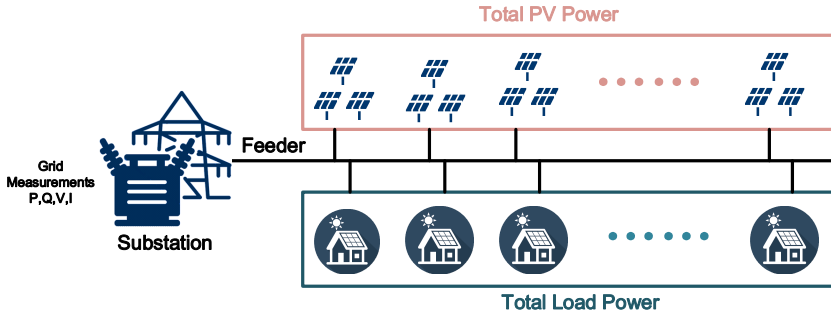


Figure 7-1. Power system with a PV system installed along the feeder.

The target of the proposed system is to decouple $P_{Net}(t)$ into $P_{Load}(t)$ and $P_{PV}(t)$, as shown in Figure 7-2. It is observed that the penetration of solar energy distorts the original demand load curve, making it difficult to recover the original demand load from the masked netload.

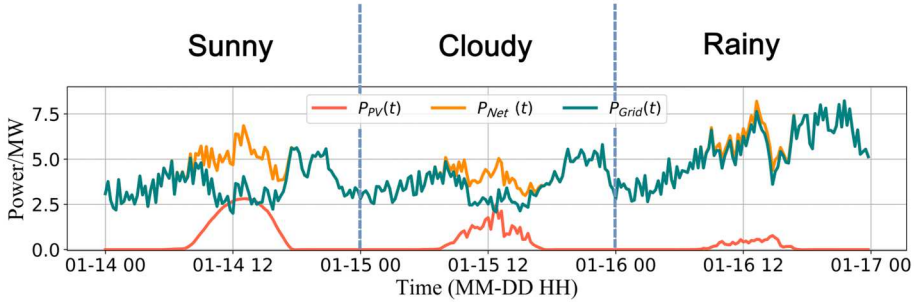


Figure 7-2. Example of time series $P_{PV}(t), P_{Net}(t), P_{Load}(t)$ under different weather conditions (Data source: Pecan Street Dataport [117]).

7.1.3 Chapter contributions

This chapter proposes a solar energy disaggregation system that enables both online and offline modes to overcome the drawbacks of existing approaches. Instead of installing measurement devices at each house, the proposed method only utilizes measurements from one smart installed in the feeder or substation to estimate the solar energy generation in the entire area. Other relevant variables, such as meteorological, irradiance, and temporal data, are also collected as the inputs of the disaggregation system. The main novelties of this chapter are as follows:

1) A CNN-BLSTM-based solar energy disaggregation system that enables online and offline modes is established. Instead of installing an electricity meter for each rooftop PV, the proposed system can decouple the PV generation and real demand load of a geographical area by installing only a feeder/substation-level smart meter or SCADA system.

2) A case study demonstrates that the proposed disaggregation system achieves high accuracy under different solar energy penetration rates and feeder models.

3) A transductive transfer learning approach that utilizes synthetic data to evaluate real-time generation at other substations or feeders is established.

7.1.4 Organization of the chapter

The remaining chapter is organized as follows: The datasets and feature extraction are introduced in Section 7.2. In section 7.3, three solar energy decoupling methods are proposed. The case studies are presented in Section 7.4. The conclusion and final discussion are drawn in the last section.

7.2 Data description

7.2.1 Feeder-level measurement

In this chapter, feeder models R1-12.47-4, R2-25.00-1, R4-25.00-1provided by GridLAB-D are employed as the model for the case study [326], see Table 7-1. The feeder types are classified depending on the residential description, ranging from light rural to moderate urban (with apparent power from 948 kW to 17021 kW). The feeder level measurement is available from physical aggregation equipment such as DSCADA, smart feeder meter, or the informatic aggregator, as introduced in Chapter 4. In this chapter, active power P_{grid}, reactive power Q_{grid} are chosen as the grid measurement variables. The duration of peak PV generation (10 am to 3 pm) and peak load is different (7 am to 10 am and 5 pm to 10 pm).

Table 7-1. Summary of prototypical feeders used in the chapter [326].

Feeder	Rated voltage/kV	Rated power/kW	Description
R1-12.47-4	12.47	5334	Heavy suburban
R2-25.00-1	24.9	17021	Moderate urban
R4-25.00-1	24.9	948	Light rural

7.2.2 Load and PV dataset

The consumer-level smart meter data is employed to construct the feeder model. In this chapter, two datasets are used for model training and testing, which are Dataport and System Advisor Model (SAM) simulation data:

Dataset 1 (Dataport). Dataport [117] is the dataset to train the proposed model. The household-level measurements are added together to construct a synthetic feeder model. In this chapter, Dataport dataset with an interval of 15 minutes is employed; 75 houses are aggregated to build a feeder with a capacity of 100 kW during Jan 2018 and Dec 2018 in Austin, Texas, US. The PV penetration rate of the feeder is controlled by limiting the percentages of houses with PV installed.

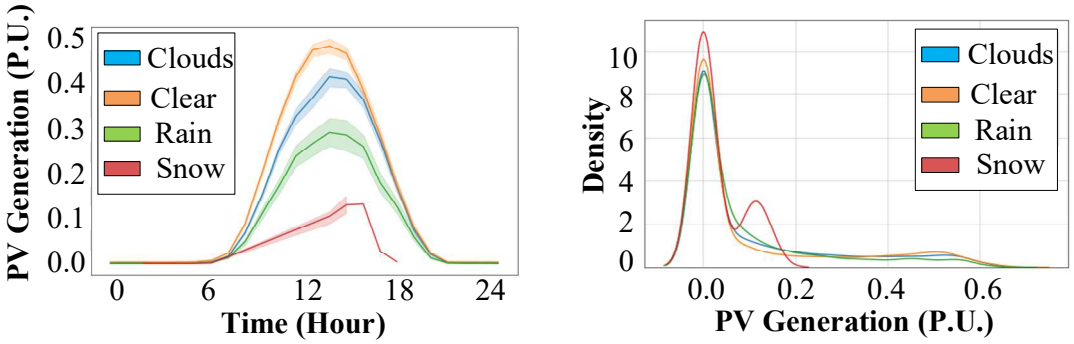
Dataset 2 (SAM simulation data). For areas where historical PV outputs are not available, a synthetic data generation approach introduced in [191] can generate

training data. SAM [384] is a techno-economic software developed by The National Renewable Energy Laboratory (NREL). The software can simulate the distribution network by integrating the renewable energy system given the geographical location (longitude and latitude) and historical weather dataset. The approach can generate the netload of the distribution network with detailed PV and combines the synthetic PV outputs with historical demand load data to simulate the feeder with solar energy penetrated.

In this chapter, both the supervised/unsupervised learning models for solar decoupling are constructed. As introduced in Sections 3.3.1 and 3.3.2, for the unsupervised learning method, the model does not have to be pre-trained with training data; only the testing dataset is needed. Moreover, only the netload measured by the smart feeder meter/DSCADA or the aggregated data from the aggregator is required, while the PV generation is not required. As for the supervised learning method, such as GBRT and deep learning models, the netload (input vector) and PV generation (label) under the feeder are required at the training stage. For supervised learning methods, the dataset is split into a training dataset (90%) and a testing dataset (10%).

7.2.3 Meteorological dataset

The 1-hourly meteorological dataset comes from National Centres for Environmental Information (NCEI) (US) [193], given that specific locations and durations are adopted. The variables include temperature T, humidity U, weather conditions (e.g., sunny, rainy, snowy, and cloudy) W, cloud cover rate D, surface albedo, pressure, wind speed, etc. Figure 7-3 compares the average PV output under different weather conditions. It is observed that the output power reaches a maximum during clear days, and less power is produced during bad weather conditions such as rainy and snowy conditions. Hence, the weather condition is also a vital variable in this case.



(a) (b)

Figure 7-3. (a) PV output under different weather conditions; (b) probability density distributions under different weather conditions (Data source: Pecan Street Dataport [117]).

7.2.4 Satellite-driven irradiance dataset

The 1-hourly irradiance measurements at the same location are obtained from the National Climatic Data Center (NCDC) [385]. The satellite-driven data include GHI, DNI, DHI, latitude, longitude, etc.

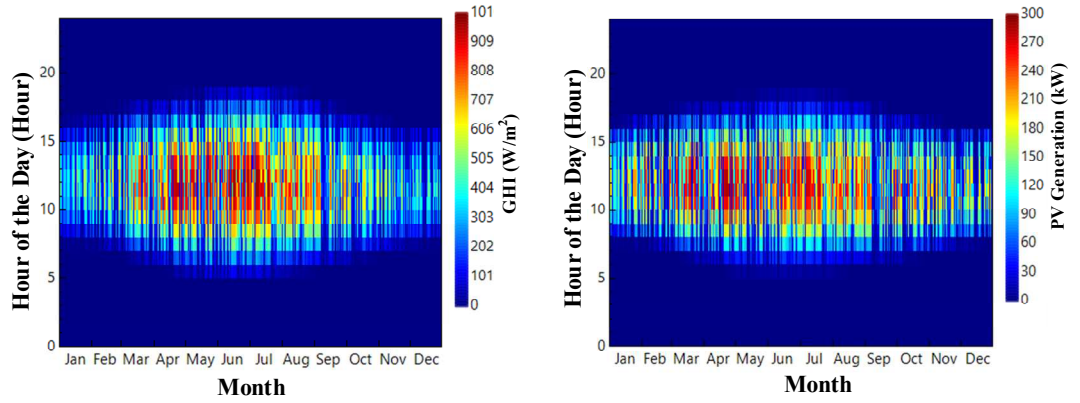
GHI: The total amount of shortwave radiation received from above by a surface horizontal to the ground.

$$GHI = DNIcos(\theta) + DHI \tag{7-2}$$

DNI: Amount of solar radiation received per unit area by a surface always held perpendicular (or normal) to the rays that come in a straight line from the sun's direction at its current position in the sky.

DHI: The amount of radiation received per unit area by a surface (not subject to any shade or shadow) that does not arrive on a direct path from the sun but has been scattered by molecules and particles in the atmosphere and comes equally from all directions.

Figure 7-4 shows the heatmap of GHI and PV output throughout the year. From the figure, it is found that the PV output is strongly correlated to the GHI value. The duration of the peak values of GHI/PV output almost overlaps.

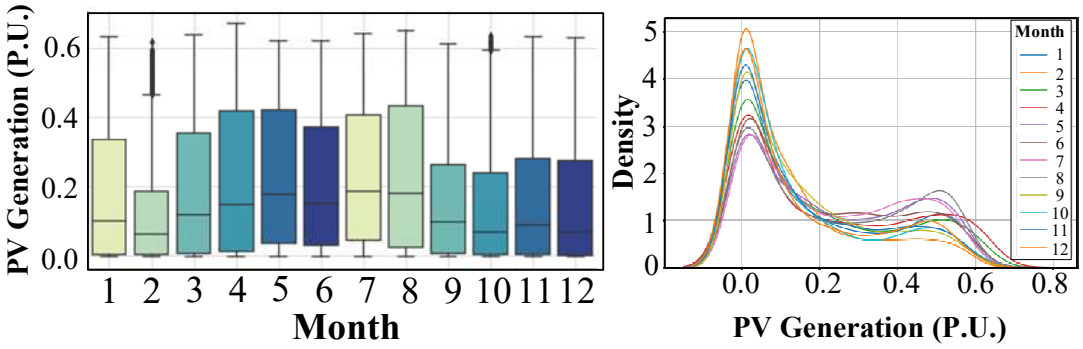


(a) GHI (b) PV Generation

Figure 7-4. Heatmap of (a) GHI/(b) PV output throughout the entire year (Data source: NCDC [385]).

7.2.5 Temporal-related features

The temporal variables include the number of hours in a day H, the month of the year M, and the quarter of the year R. Examples of average PV outputs and probability density distributions during different months are presented in Figure 7-5. It is observed that both the month and the quarter of the year influence PV output. Normally, the maximum output throughout the year appears between June and August.



(a) (b)

Figure 7-5. (a) Bar chart of PV outputs in different months (b) probability density distributions during different months (Data source: Pecan Street Dataport [117]).

7.2.6 Data Preparation

The data collected from various resources should be processed before sending it to the solar energy decoupling model. The data preparation process includes cleaning, synchronization, and one-hot encoding.

1) **Data cleaning**: The purpose of data cleaning is to generate clean and structured data. As introduced in section 3.2.1, the original data contains high-frequency noise, missing values, wrong labels, and duplicates. The load data is filtered by a two-level DWT denoising filter proposed in Section 6.2.3 in Chapter 6, and the neighbouring values fill the missing values.
2) **Data Synchronization**: It should be noticed that the different datasets measured with various sampling rates (15 minutes for the load, 1 hour for weather and irradiance data), so the data from all datasets should be synchronized. In this work, all data is aligned using the timestamps from each source and resamples the data interval to 15 minutes.
3) **One-hot encoding:** Before feeding the data to the DNN model, all categorical variables should be converted to numerical forms via one-hot encoding. A new binary variable represents the original variable [240, 243]. In this work, the categorical variable matrix $\boldsymbol{C_t}$ contains:

$$\boldsymbol{C_t} = [W_t\ , H_t\ , M_t, R_t] \tag{7-3}$$

By implementing one-hot encoding, the variables are transformed to:

$$\boldsymbol{C_t^O} = f^O(\boldsymbol{C_t}) \tag{7-4}$$

where f^O is the one-hot encoding function, and $\boldsymbol{C_t^O}$ is the one-hot encoding matrix. Hence, the overall input matrix $\boldsymbol{D}$ is shown as follows:

$$\boldsymbol{D_t} = [\boldsymbol{N_t}, \boldsymbol{C_t^O}] \tag{7-5}$$

$$\boldsymbol{N_t} = [P_{grid,t}\ , Q_{grid,t}, T_t\ , U_t\ , D_t, GHI_t, DNI_t, DHI_t] \tag{7-6}$$

where $\boldsymbol{N_t}$ is the numerical variable.

7.3 Behind-the-meter solar energy detection – three methods

In this work, three solar energy detection algorithms are proposed: an unsupervised Upscaling Method (UM), supervised Gradients Boosting Regression Tree (GBRT)-based algorithms and the deep learning method.

7.3.1 Method I: Unsupervised upscaling method

As shown in Figure 7-6 (a), the PV generation highly correlates with the ambient temperature T and solar irradiance. More solar energy is generated given larger GHI and higher T. Moreover, Figure 7-6 (b) plots the PV generation and GHI in one week together. The figure shows that the shape of the PV generation curve is highly overlapping with the curve of GHI in the same area. The unsupervised learning approach utilizes real-time GHI measurement and historical feeder measurements only to estimate the PV outputs.

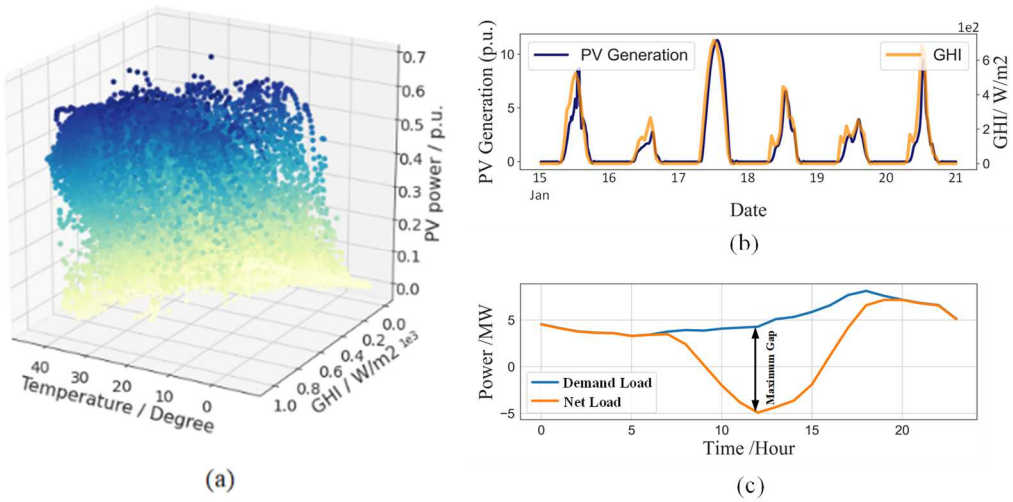


Figure 7-6. (a) The 3D plot of the combined effect of temperature and GHI on PV output (b) Comparison of PV output and GHI; (c) Comparison of net load and demand load.

7.3.1.1 Estimate PV capacity by edge detection

The PV capacity C under the feeder is first estimated via an edge detection method. Assuming the load demand under the feeder keeps stable, given historical feeder demand before PVs installed $P_{without\ PV,t}$ and feeder demand after PVs installed $P_{with\ PV,t}$, the mismatching between $P_{without\ PV,t}$ and $P_{with\ PV,t}$ can be calculated by (7-7):

$$Error_t = P_{without\ PV,t} - P_{with\ PV,t} \tag{7-7}$$

The PV capacity C is equal to the maximum of $Error$ throughout the whole year approximately:

$$C \approx max\ (Error_t) \tag{7-8}$$

7.3.1.2 Estimate PV output

The PV output is estimated via normalized GHI and PV capacity C:

$$P_{PV,t} = C \cdot GHI_t \tag{7-9}$$

The unsupervised method is easy to implement, does not require the model to be trained, and only a few measurements are needed. This method is highly suitable for areas that lack smart meters. However, this method does not consider other relevant variables, such as temperature, and cloud cover rate, so the model cannot provide an exact estimation.

7.3.2 Method II: Supervised Gradient Boosting Regression Tree-based method

The supervised GBRT -based solar energy detection algorithm requires training machine learning models ahead. GHI and feeder demand measurements are adopted as input features.

The GBRT algorithm's core components are a machine learning algorithm that produces a prediction model from a series of weak prediction models [256]. Usually, the GBRT algorithm contains three elements: a differentiable loss function for optimization, a squared error is adopted as the loss function for regression task; a weak prediction model to make a prediction, and a decision tree is used as the weak model in GBRT; and an additive model that can add all weak models together and minimize the losses, see Algorithm 3-1 in Chapter 3.

7.3.3 Method III: Deep learning model

Apart from the unsupervised/supervised machine learning models introduced above, high computational ability deep learning models are also developed, detailed introduction of the deep learning-based solar energy decoupling model is developed as follows.

The proposed model is assumed to have the authority to access the aggregated data from the physical/informatic aggregator. In addition, the model can also obtain measurements from weather stations and satellites. The system's target is to separate the net load into the demand load and PV generation. The proposed system enables two operating modes: offline training and online learning, as shown in Figure 7-7. The model is trained with historical data at first, and the online mode can provide PV disaggregation on a real-time basis.

1) Offline training mode: A supervised learning method, the model, should be pre-trained with historical netload and PV generation data offline. Since the historical PV generation data is not always available for the energy utility, there are several approaches to obtain such label training data: (1) Utilize the public dataset at the research location, such as Dataport [117] for the distribution network in Texas, US. Such a public dataset is anonymized and under the permission of consumers. (2) Utilize software simulation software to generate synthetic netload and PV generation data. (3) Utilize the transfer learning method as introduced in Section 7.4.4. (4) Select

a small group of volunteers under the distribution network, and the energy utility collects the smart meter data and the PV generations inside the volunteers' houses under their permission. Since the trials are already under the volunteers' consent, such trials will not cause privacy issues.

1) Online Mode: The online mode part of the system consists of three components: real-time measurement, a cloud server, and power utility. In the online mode, the PV disaggregation system can access real-time measurements from distribution feeders and weather stations/satellite systems. The grid measurements include active power, reactive power, voltage, current, etc. The weather-related measurements include temperature, humidity, cloud cover, etc. After data are gathered from the sensors/stations, the data are fed into the core part of the system, the cloud platform, which implements deep learning algorithms (introduced in the next subsection) to decouple the original net load into PV generation and the real demand load.

Conventional BLSTM cannot support online learning due to the delay problem. This is because, during the online mode, it is assumed that the length of the input sequence is unknown, and it is impossible to learn the input sequence from both forward and backward directions. Hence, an online BLSTM algorithm originally used for online speech recognition is adopted [386]. A sliding window moves over the real-time input sequence, and then the BLSTM is implemented for each sliding window. In this case, the model can learn bidirectionally, and the time delay is reduced to T_w. The T_w sliding window moves at time step T_s. Therefore, the original online sequence can be split into a few chunks, and the ith window is:

$$\boldsymbol{Z}_i = [z_{iT_s+1}, z_{iT_s+2}, \dots, z_{iT_s+T_W}] \tag{7-10}$$

and a maximum number of T_w/T_s windows overlap at time stamp t. The final output at time stamp t is evaluated by averaging the output of overlapping windows at t. Note that the online system results in a delay of T_w since the system should lookup timestamp $T_w - 1$ in the future to determine the output at time stamp t.

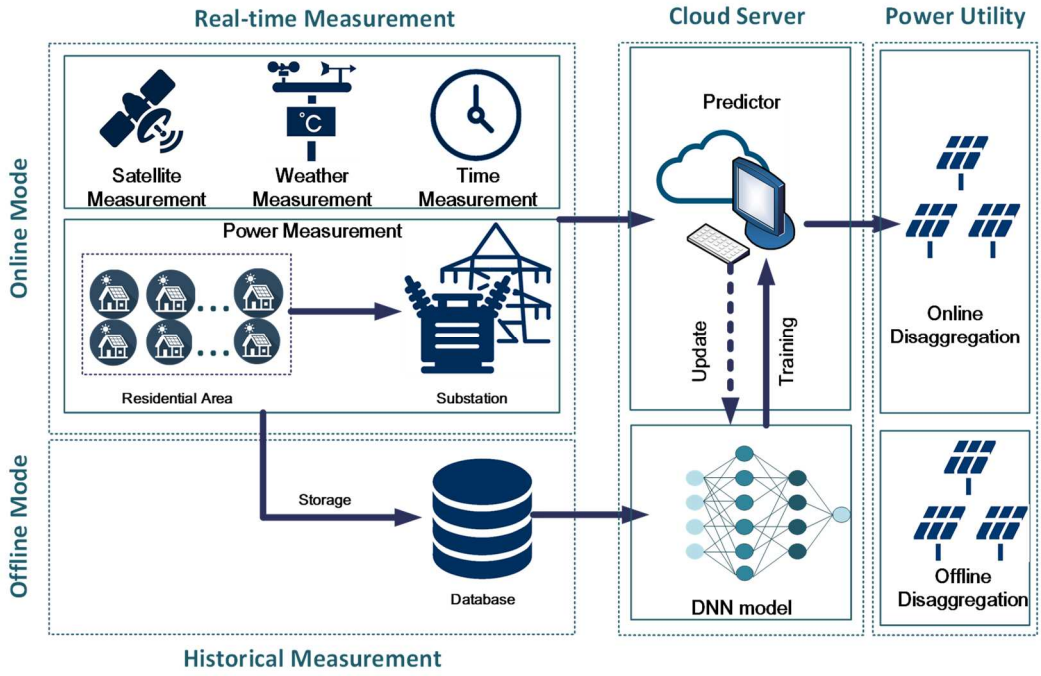


Figure 7-7. Online/Offline PV energy disaggregation framework.

1D CNN-BLSTM combines a BLSTM RNN with a one-dimensional CNN and provides a deeper learning ability for regression tasks with time-series data [243]. 1D CNN model is efficient in extracting the important features from the 1D time-series data and can filter out the noise of the input data. However, the 1D CNN model only focuses on the local trend and has limited ability to adapt to long temporal dependencies [387]. In contrast, the LSTM model performs better in tracking long-term dependencies from the input sequence, but the ability to extract local features is limited. Hence, a model that combines the 1D CNN model and LSTM model is expected to take advantage of these two deep learning techniques and achieve higher prediction accuracy [388]. The structure of the 1D CNN-BLSTM network utilized in this work is presented in Figure 7-8; the network contains six layers, which are one input layer, one 1D CNN layer, one max-pooling layer, one BLSTM layer, one fully connected layer, and one output layer:

1. **Input Layer:** The network's input is the multivariate dataset which contains the features from four datasets, as illustrated in Figure 7-7.

2. **1D Convolutional Layer:** The input layer is followed by two 1D convolutional layers with 64 and 32 filters, respectively. The kennel size of the first 1D convolutional layer is 5 with strides=1, and the padding type is set as "causal"; the kennel size of the second 1D convolutional layer is 3 with strides=1 and the padding type is set as "causal". In addition, each convolutional layer is linked with a 1D max-pooling layer. The function of the max-pooling is to calculate the maximum value in each patch of each feature map. In this model, a 1D max-pooling layer with kernel size two and stride one was constructed.
3. **BLSTM Layer:** Two BLSTM layers with 256 units are stacked to enable the long-term temporal dependencies on the feature extracted by the convolutional layer. Each BLSTM layer contains two LSTM layers of opposite directions to the same output. This structure enables the output layer to learn both forward and backward information.
4. **Flatten Layer:** After BLSTM layers, a flatten layer is employed to flatten the 3×512 matrix into a vector with a size of 1536. The flatten layer is normally used in the transition from LSTM or convolutional layer to make the multi-dimensional input one-dimensional.
5. **Fully Connected Layer:** The flatten layer is then connected with two fully connected layers with 64 and 16 nodes, respectively. Two dropout layers with rates 0.2 and 0.1 are stacked after each fully connected layer to avoid overfitting problems during the training process.
6. **Output Layer:** The final fully connected layer is connected to the output layer with two nodes which estimate the PV generation and the demand load, respectively.

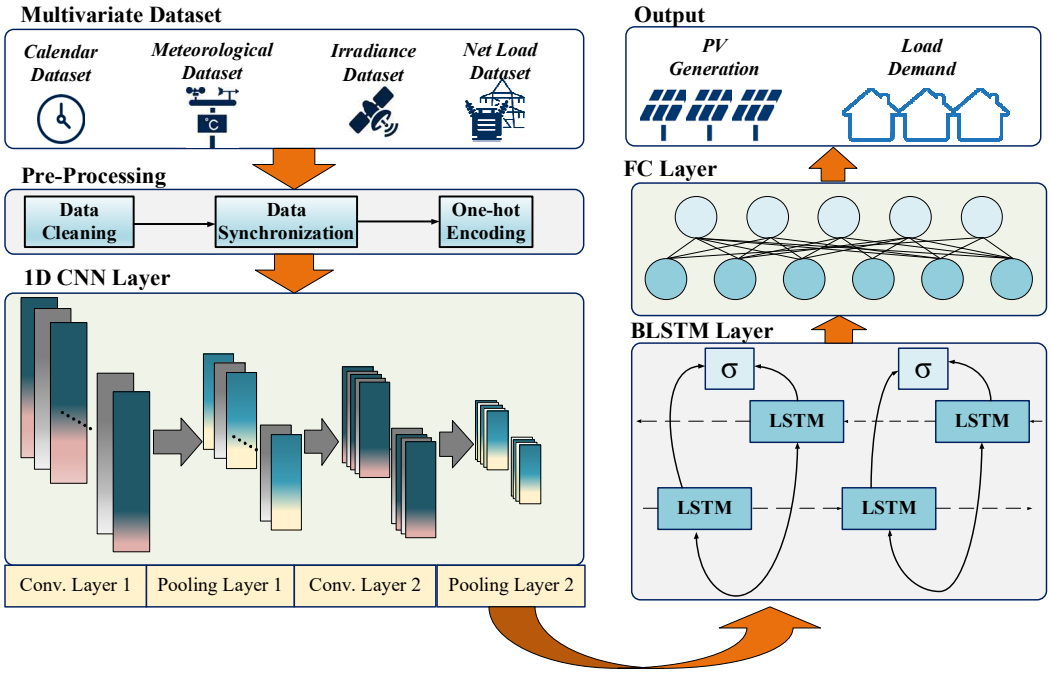


Figure 7-8. The structure of the proposed 1D CNN-BLSTM network.

The model complexity (O) is determined by computing the parameters of each layer. Referring to [389], for the 1D CNN layer, the computation complexity for each sample is computed by Eqn. (7-11):

$$O(1D\ CNN) = O\left(\sum_{k=1}^{K_C} (L_{c,k} \times I_{c,k} + 1) \times F_{c,k}\right) \tag{7-11}$$

where K_c is the number of the convolutional layers, $F_{c,k}$ indicates the number of filters in the kth layer, $L_{c,k}$ is the kernel length, and $I_{c,k}$ denotes the number of input channels. As for the BiLSTM layer, the computation complexity is:

$$O(LSTM) = O\left(\sum_{k=1}^{K_B} 2 \times 4 \times \left(\left(I_{B,k} + 1\right) \times M_{B,k} + {M_{B,k}}^2\right)\right) \tag{7-12}$$

where K_L is the number of the BiLSTM layers, $I_{B,k}$ is the number of input channels, $M_{B,k}$ denotes the number of LSTM units. In turns of the FC layer, the computation complexity is computed as:

$$O(FC) = O\left(\sum_{k=1}^{K_F} ((N^{in}{}_{F,k} + 1) \times N^{out}{}_{F,k})\right) \tag{7-13}$$

where K_L is the number of FC layers, ${N^{in}}_{F,k}$ is the number of the input neurons, and ${N^{out}}_{F,k}$ denotes the number of the input neurons. Hence, the overall computation complexity of the 1D CNN-BLSTM model is:

$$\begin{aligned} O &= O(1D\ CNN) + O(LSTM) + O(FC) \\ &= O\left(\sum_{k=1}^{K_C} (L_{c,k} \times I_{c,k} + 1) \times F_{c,k}\right) + O\left(\sum_{k=1}^{K_B} 2 \times 4 \times \left(\left(I_{B,k} + 1\right) \times M_{B,k} + {M_{B,k}}^2\right)\right) + \\ &\quad O\left(\sum_{k=1}^{K_F} ((N^{in}{}_{F,k} + 1) \times N^{out}{}_{F,k})\right) \end{aligned} \tag{7-14}$$

In this work, K_C equals to 2, K_B is chosen as 1, and K_F is 2. Based on the computation complexity in Eqn. (7-14), the model parameters of the proposed 1D CNN-BLSTM model are summarized in Table 7-2; both the type of hyperparameter, shape, and number of parameters are concluded. The total parameters of the proposed model are 2,285,906, and 12.5 MB (32-bit floats) is required to store all parameters. The DNO will implement the solar energy disaggregation, which utilizes Energy Management Systems (EMS) to operate the proposed solar energy disaggregation model [390]. EMS is equipped with a high computational machine that enables various machine learning/deep learning-based applications.

Table 7-2. Model parameters of the proposed 1D CNN-BLSTM model.

Layer	Hyperparameters Setting	Output Shape	Num. of Parameters
Conv1D_1	filters=64, kernel size=5, strides=1, padding="causal"	5, 64	13504
MaxPooling1D_1	pool size=2, strides=1, padding="valid"	4, 64	0
Conv1D_2	filters=32, kernel size=3, strides=1, padding="causal"	4, 32	6176
MaxPooling1D_2	pool size=2, strides=1, padding="valid"	3, 32	0
BiLSTM_1	256, return sequences=True	3, 512	591872
BiLSTM_2	256, return sequences=True	3, 512	1574912
Flatten	-	1536	0
Dense_1	64, activation="relu"	64	98368
Dropout_1	0.2	64	0
Dense_2	16, activation="relu"	16	1040
Dropout_2	0.1	16	0
Dense_3	2, activation="sigmoid"	2	34

Total params: 2,285,906; Trainable params: 2,285,906; Memory size: 12.5 MB.

7.4 Results and Discussion

In this section, three case studies are simulated with the previous two datasets. Both the unsupervised learning model, supervised machine learning model, and supervised deep learning models are evaluated. Furthermore, a transfer learning approach is proposed to assess the transferability of the proposed solar decoupling method.

7.4.1 Performance Evaluation

7.4.1.1 Software& hardware

The simulation and computations are conducted on a Dell laptop equipped with a Core i7–7700HQ CPU, an NVIDIA GTX 1060 GPU, and 8 GB RAM. The deep learning algorithm runs on Python 3.6, and the TensorFlow framework is adopted to train the DNN model.

7.4.1.2 Experimental setup

In this section, three case studies are used to investigate how the penetration rate and hyperparameters influence the performance of the proposed disaggregation algorithms. Moreover, data transferability is studied to investigate whether a model can be trained with a synthetic dataset.

7.4.1.3 Evaluation criteria

To evaluate the performance of proposed disaggregation algorithms, three evaluation metrics are adopted, which are RMSE, nRMSE, and R^2.

7.4.2 Case study I: Comparison between supervised and unsupervised machine learning methods

The case study evaluates both the performance of supervised and unsupervised solar energy separation methods. Given the net load measured at the feeder side, the solar energy separator aims to estimate the PV generation in real-time. Figure 7-9 presents a comprehensive analysis of the two algorithms. The estimating curves evaluated by two models and the ground truth PV generation are shown in Figure 7-9 (c). The actual value is shown in light blue shading, while the solid red curve and the solid orange curve represent the estimating results from GBRT and UM models, respectively. From the figure, the values estimated by the GBRT method track the ground truth values with high accuracy, while the UM method cannot estimate the peak values generated by the PV. The estimation evaluation metrics of the two methods are shown in the radar chart (Figure 7-9 (a)) and Table 7-3, and the best metrics are highlighted with grey shading. It is observed that the GBRT method is superior to UM method in all metrics. The nRMSE values of UM and GBRT are12.41% and 4.68%, while the RMSE values of UM and GBRT methods are 1.54 MW and 0.58 MW, respectively.

Moreover, the nMAE values of UM and GBRT methods are 6.44% and 2.55%. Meanwhile, the correlation metrics, R2 and ρ, provide a more detailed view than the GBRT method is much better than UM method. Figure 7-9 (b) utilizes a scatter plot to visualize the correlation of estimated values with the ground truth values. R2 and ρ of the GBRT method reach 96% and 95%, which means the estimating is highly correlated with the actual values, while R2 and ρ of UM are only 73% and 54%. Although the GBRT method has superior performance, it requires pretraining the model before adopting it to real-time applications. Meanwhile, UM has lower accuracy but high flexibility.

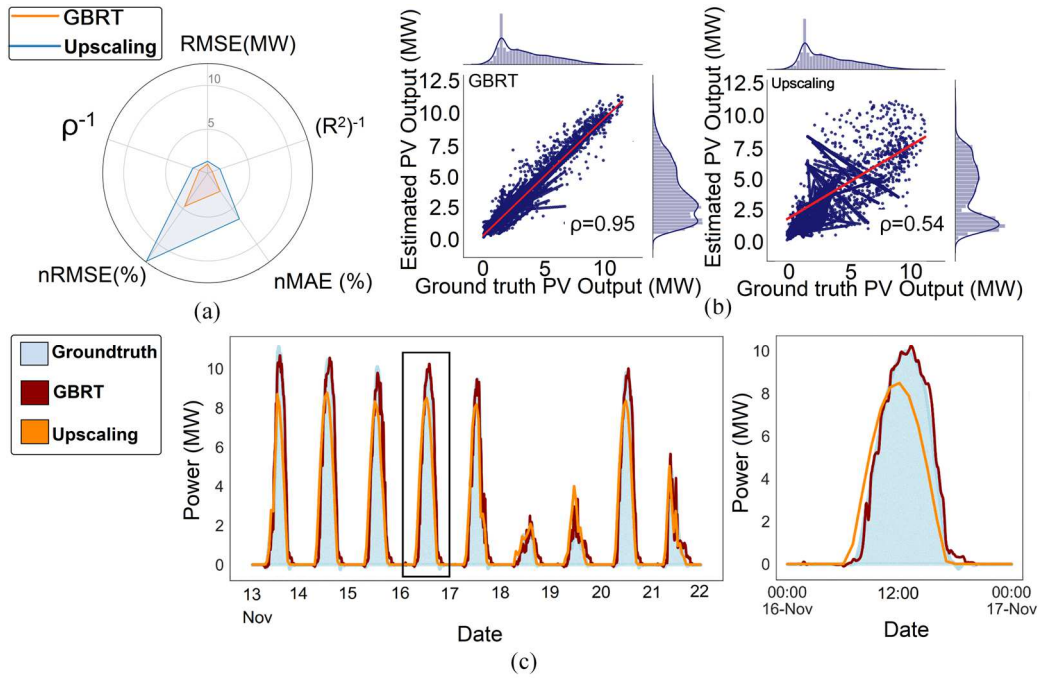


Figure 7-9. (a) Radar chart of performance metrics to two PV separation algorithms; (b) Scatter plot of estimated PV power versus ground truth PV energy for unsupervised upscaling and gradient boosting methods, with the Pearson correlation. (b) Comparison of solar energy estimated by the PV separator and ground truth value.

Table 7-3. Performance of unsupervised/supervised solar energy decoupling methods.

Algorithms	nRMSE (%)	RMSE (MW)	nMAE (%)	R^2	ρ
Unsupervised Algorithm	12.41	1.54	6.44	0.73	0.54
GBRT Algorithm	4.68	0.58	2.55	0.96	0.95

7.4.3 Case study II: Performance of deep learning models under different PV penetration rates

In this case study, the proposed 1D CNN-BLSTM solar energy disaggregation model is compared to state-of-the-art solar energy disaggregation models. Three feeder models (the light rural feeder model with $\overline{P}_L$ =948 kW, heavy suburban feeder model with $\overline{P}_L$ =5335 kW, and the moderate urban feeder model with $\overline{P}_L$ =17021 kW) with different PV penetration rates (5%, 10%, 20%, 30%, 40%, 50%) described in Table 7-1 are studied. Regarding the net load, the PV penetration rate is defined as the percentage of PV output compared to the peak demand load:

$$PV\ peneration\ (\alpha) = \frac{Peak\ PV\ Load}{peak\ demand\ load} \quad (7\text{-}15)$$

The accuracies and the required computation time of the estimated solar energy by different models are shown in Tables 7-4 to 7-6. In the table, all models can be divided into three groups: model-based method, upscaling method, and data-driven methods. Whist data-driven methods are further classified into machine learning-based and deep learning-based models. The tables show the superiority of the deep learning-based models over other models in terms of estimation error and accuracy. In contrast, the deep learning models require much longer computation time and larger memory space. From Tables 7-4 to 7-6, the model-based model, which constructs the mathematical models referring to the configuration of PV panels and meteorological variables, shows a lower accuracy among all models. In addition, the machine learning methods, including KNN regression, SVM regression, RF regression, and GBRT, achieve better estimation results than model-based and upscaling methods. However, conventional machine learning models cannot capture high-dimensional nonlinear patterns from the input netload, which limits the accuracy of the models. When comparing the five deep learning-based models, the naive model, MLP, performs the worst in all cases; this result is due to the MLP model cannot capture the long-term temporal dependencies.

In contrast, the memory cells inside the LSTM/GRU models make the models computationally more efficient. Furthermore, the 1D CNN model achieves similar accuracy as the LSTM/GRU models, and the 1D convolutional layers help the model extract the complex pattern from the 1D time-series data. The proposed 1D CNN-BLSTM takes advantage of the LSTM and 1D CNN models to extract the local pattern and capture the long-term dependencies simultaneously. From Tables 7-4 to 7-6, it is observed that the proposed 1D CNN-BLSTM performs the best among all models for all cases. For instance, the 1D CNN-BLSTM improves the MAE and nRMSE of PV estimation by 55.89% and 53.74% compared to the conventional model-based model when α=5% and $\overline{P}_L$ =948 kW.

Table 7-4. Disaggregation performance under different penetration rates (light rural feeder model).

α (%)	Metrics	Data-Driven Method								
		Machine Learning Models				Deep Learning Models				
		SVM+PCA	KNN+PCA	RF+PCA	GBRT	CNN	GRU	LSTM	MLP	CNN-BiLSTM
5	R^2	0.73	0.64	0.78	0.77	0.78	0.79	0.83	0.74	**0.80**
	MAE (kW)	4.16	3.42	2.56	3.02	3.30	2.30	1.79	2.92	**1.64**
	nRMSE (%)	0.65	0.75	0.58	0.60	0.54	0.50	0.37	0.57	**0.35**
	RMSE (kW)	6.19	7.20	5.56	5.75	5.20	4.77	3.58	5.40	**3.36**
	Computation Time (s)	14.77	35.16	60.30	21.91	45.61	305.21	310.81	51.90	**155.20**
10	R^2	0.73	0.63	0.80	0.78	0.80	0.82	0.83	0.77	**0.87**
	MAE (kW)	8.44	6.95	4.95	5.97	6.62	4.25	4.92	5.66	**2.96**
	nRMSE (%)	1.30	1.53	1.11	1.19	1.21	0.78	1.00	1.12	**0.68**
	RMSE (kW)	12.38	14.55	10.58	11.29	11.52	7.44	9.52	10.70	**6.51**
	Computation Time (s)	14.26	29.23	60.83	23.25	44.77	311.45	312.24	52.34	**158.12**
20	R^2	0.74	0.63	0.82	0.79	0.80	0.80	0.85	0.79	**0.89**
	MAE (kW)	16.65	13.25	9.14	11.06	11.89	7.49	5.47	10.73	**6.01**
	nRMSE (%)	2.51	2.97	2.08	2.24	2.37	1.77	1.17	2.32	**1.23**
	RMSE (kW)	23.85	28.15	19.74	21.25	22.54	16.86	11.15	21.46	**11.71**
	Computation Time (s)	14.65	30.17	61.67	25.47	48.47	346.71	356.93	51.12	**230.85**
30	R^2	0.74	0.64	0.82	0.79	0.81	0.79	0.80	0.80	**0.93**
	MAE (kW)	23.92	19.44	13.52	16.30	18.81	13.52	17.25	17.63	**7.66**
	nRMSE (%)	3.70	4.38	3.11	3.36	3.41	2.73	2.86	3.13	**1.48**
	RMSE (kW)	35.16	41.53	29.51	31.87	32.35	25.92	27.19	29.70	**14.08**
	Computation Time (s)	15.03	28.85	62.16	22.46	44.22	322.07	341.02	20.35	**208.47**
40	R^2	0.75	0.66	0.82	0.80	0.80	0.85	0.90	0.72	**0.94**
	MAE (kW)	30.18	25.16	17.40	20.86	24.82	13.70	11.26	21.78	**10.43**
	nRMSE (%)	4.81	5.69	4.07	4.35	4.58	2.84	2.38	3.87	**2.25**
	RMSE (kW)	45.60	53.98	38.60	41.28	43.48	26.92	22.57	36.69	**21.41**
	Computation Time (s)	13.92	29.08	65.49	22.31	46.85	355.34	364.12	29.28	**370.45**
50	R^2	0.79	0.68	0.82	0.80	0.83	0.88	0.87	0.79	**0.96**
	MAE (kW)	33.02	29.66	22.00	26.13	26.37	14.82	15.93	22.91	**11.55**
	nRMSE (%)	5.51	6.77	5.16	5.34	4.60	2.98	3.17	4.14	**2.38**
	RMSE (kW)	52.31	64.24	48.95	50.66	43.65	28.28	30.13	39.25	**22.58**
	Computation Time (s)	13.95	28.84	62.38	21.84	44.77	325.63	344.22	22.51	**312.73**

Table 7-5. Disaggregation performance under different penetration rates (heavy suburban feeder model).

α (%)	Metrics	Data-Driven Method								
		Machine Learning Models				Deep Learning Models				
		SVM+PCA	KNN+PCA	RF+PCA	GBRT	CNN	GRU	LSTM	MLP	CNN-BiLSTM
5	R^2	0.73	0.63	0.78	0.77	0.72	0.79	0.78	0.81	**0.90**
	MAE (kW)	22.77	18.90	14.25	16.51	20.08	14.12	14.78	12.56	**10.04**
	nRMSE (%)	0.63	0.74	0.57	0.58	0.69	0.45	0.46	0.43	**0.37**
	RMSE (kW)	33.84	39.67	30.57	31.13	37.06	24.02	24.71	23.38	**19.06**
	Computation Time (s)	16.71	29.22	61.38	21.66	42.62	302.31	351.67	48.15	**236.00**
10	R^2	0.73	0.63	0.80	0.78	0.80	0.82	0.81	0.75	**0.89**
	MAE (kW)	44.83	36.61	27.03	31.99	30.22	18.66	20.39	27.80	**16.61**
	nRMSE (%)	1.23	1.44	1.07	1.12	1.03	0.73	0.74	0.96	**0.59**
	RMSE (kW)	65.90	76.92	57.18	59.99	55.43	39.18	39.71	51.57	**31.66**
	Computation Time (s)	15.40	29.47	61.92	22.06	39.35	345.72	311.36	44.60	**411.04**
20	R^2	0.74	0.63	0.81	0.79	0.80	0.81	0.82	0.83	**0.93**
	MAE (kW)	93.00	74.10	51.39	61.80	65.23	46.22	44.84	51.64	**33.36**
	nRMSE (%)	2.50	2.95	2.08	2.23	2.33	1.73	1.30	1.35	**1.19**
	RMSE (kW)	133.61	157.59	111.21	119.41	124.81	92.50	68.24	106.62	**63.56**
	Computation Time (s)	14.76	28.89	65.37	21.74	39.42	335.67	343.93	18.84	**347.82**
30	R^2	0.74	0.64	0.82	0.79	0.76	0.87	0.83	0.71	**0.93**
	MAE (kW)	134.89	108.64	74.50	91.26	124.34	58.19	56.23	97.92	**39.90**
	nRMSE (%)	3.70	4.36	3.06	3.33	3.77	2.16	2.26	3.20	**1.48**
	RMSE (kW)	197.56	232.79	163.62	177.86	201.15	115.36	120.88	171.16	**79.30**
	Computation Time (s)	14.82	29.00	69.26	22.13	36.73	354.08	340.42	24.17	**255.37**
40	R^2	0.75	0.66	0.83	0.79	0.80	0.87	0.81	0.80	**0.94**
	MAE (kW)	171.16	138.03	97.04	117.86	139.97	64.85	80.77	112.46	**52.34**
	nRMSE (%)	4.81	5.57	4.00	4.35	4.15	2.49	3.16	3.43	**1.83**
	RMSE (kW)	256.84	297.65	213.66	232.40	221.59	133.18	169.00	183.22	**97.78**
	Computation Time (s)	14.96	29.00	72.90	22.03	31.74	353.76	387.92	19.35	**336.98**
50	R^2	0.78	0.69	0.82	0.81	0.83	0.91	0.91	0.85	**0.94**
	MAE (kW)	188.26	164.32	122.20	145.93	136.30	73.97	80.05	115.39	**61.71**
	nRMSE (%)	5.60	6.70	5.16	5.29	4.41	2.65	2.76	3.68	**2.26**
	RMSE (kW)	298.93	357.81	275.58	282.69	235.64	141.46	147.46	196.47	**121.05**
	Computation Time (s)	13.65	29.98	67.27	22.55	39.21	301.14	322.76	33.96	**218.20**

Table 7-6. Disaggregation performance under different penetration rates (moderate urban feeder model).

α (%)	Metrics	Data-Driven Method								
		Machine Learning Models				Deep Learning Models				
		SVM+PCA	KNN+PCA	RF+PCA	GBRT	CNN	GRU	LSTM	MLP	CNN-BiLSTM
5	R^2	0.74	0.64	0.79	0.77	0.84	0.81	0.82	0.79	**0.85**
	MAE (kW)	70.97	59.11	44.03	51.81	48.19	54.91	40.82	49.59	**28.66**
	nRMSE (%)	0.61	0.72	0.55	0.57	0.48	0.62	0.50	0.53	**0.35**
	RMSE (kW)	105.50	123.81	94.47	98.09	83.12	106.64	86.04	91.50	**60.14**
	Computation Time (s)	15.02	29.71	66.97	20.93	31.77	345.92	342.34	42.82	**472.88**
10	R^2	0.74	0.63	0.80	0.79	0.80	0.84	0.83	0.81	**0.87**
	MAE (kW)	146.00	119.08	86.21	101.49	91.28	73.51	78.06	82.82	**53.23**
	nRMSE (%)	1.24	1.47	1.08	1.12	0.94	0.84	0.78	0.93	**0.64**
	RMSE (kW)	212.49	250.47	184.76	191.07	160.93	143.24	133.93	158.36	**110.40**
	Computation Time (s)	14.94	30.38	66.08	20.73	38.89	352.83	353.12	38.92	**453.87**
20	R^2	0.73	0.63	0.81	0.80	0.82	0.85	0.85	0.77	**0.91**
	MAE (kW)	292.52	234.46	165.10	188.69	199.87	112.84	126.06	200.29	**93.31**
	nRMSE (%)	2.47	2.92	2.09	2.23	1.95	1.38	1.64	2.13	**1.06**
	RMSE (kW)	421.47	497.10	357.16	369.37	333.24	235.87	279.75	362.55	**180.54**
	Computation Time (s)	15.44	29.32	66.35	20.66	33.02	311.54	305.87	28.56	**423.15**
30	R^2	0.62	0.65	0.72	0.75	0.77	0.89	0.84	0.81	**0.91**
	MAE (kW)	608.44	318.15	300.33	301.41	355.59	151.28	204.67	262.56	**185.27**
	nRMSE (%)	4.20	4.03	3.57	3.36	3.43	1.75	2.35	2.56	**2.07**
	RMSE (kW)	715.83	686.94	608.93	572.57	584.89	297.88	400.96	436.43	**353.71**
	Computation Time (s)	8.42	29.49	65.68	20.85	36.55	311.24	300.25	26.69	**229.55**
40	R^2	0.75	0.66	0.82	0.79	0.80	0.89	0.89	0.81	**0.93**
	MAE (kW)	546.49	442.87	315.37	380.10	409.03	187.40	193.35	344.51	**211.77**
	nRMSE (%)	4.82	5.60	4.06	4.34	3.85	2.32	2.33	3.46	**2.35**
	RMSE (kW)	820.51	954.17	692.74	740.37	656.03	395.13	397.88	589.87	**400.27**
	Computation Time (s)	14.54	29.15	62.21	20.81	34.25	341.59	334.24	36.25	**251.96**
50	R^2	0.79	0.69	0.82	0.81	0.82	0.84	0.86	0.82	**0.94**
	MAE (kW)	589.91	527.93	388.02	461.86	500.72	331.88	330.04	373.61	**284.39**
	nRMSE (%)	5.53	6.73	5.12	5.28	4.81	3.64	3.76	3.84	**2.90**
	RMSE (kW)	941.40	1146.06	872.95	899.08	819.51	621.09	641.25	654.14	**493.71**
	Computation Time (s)	14.11	29.19	66.18	22.35	36.80	309.23	322.76	24.78	**431.34**

The estimation results of the PV generation and the demand load are visualized in Figures 7-10 and 7-11, and these two figures show the cases of $\bar{P}_L$ =948 kW, α=20% and $\bar{P}_L$ =17021 kW and α=5% cases, respectively. In addition, the results estimated by GBRT, LSTM, and MLP are shown in the same figure to compare to the proposed 1D CNN-BLSTM model. The original net load curve is presented in the top plot of

Figure 7-10. Compared to the net load without PV systems, a dramatic drop is observed between 8:00 am and 5:00 pm, a typical characteristic of identifying PV generation. The middle plot shows the estimations of the PV outputs performed by the four algorithms, and the bottom plot shows the decoupled demand load curve. The figure shows that although the PV outputs inferred by all algorithms closely match the ground truth curve, the proposed CNN-BLSTM presents the best estimation. In this case, the R^2 for CNN-BLSTM is approximately 0.89, and the nRMSE is 1.23%. In turns of Figure 7-11, a case with a low PV penetration rate (5%), the figure shows that although the benchmark models (GBRT, LSTM and MLP) can estimate the PV generation with high accuracy during the time with abundant sunlight, e.g., sunny days, the accuracy is reduced in bad weather conditions such as rainy and cloudy days. However, the proposed model can make a great estimation in almost all-weather conditions, demonstrating the improvement of the proposed model compared to the existing works.

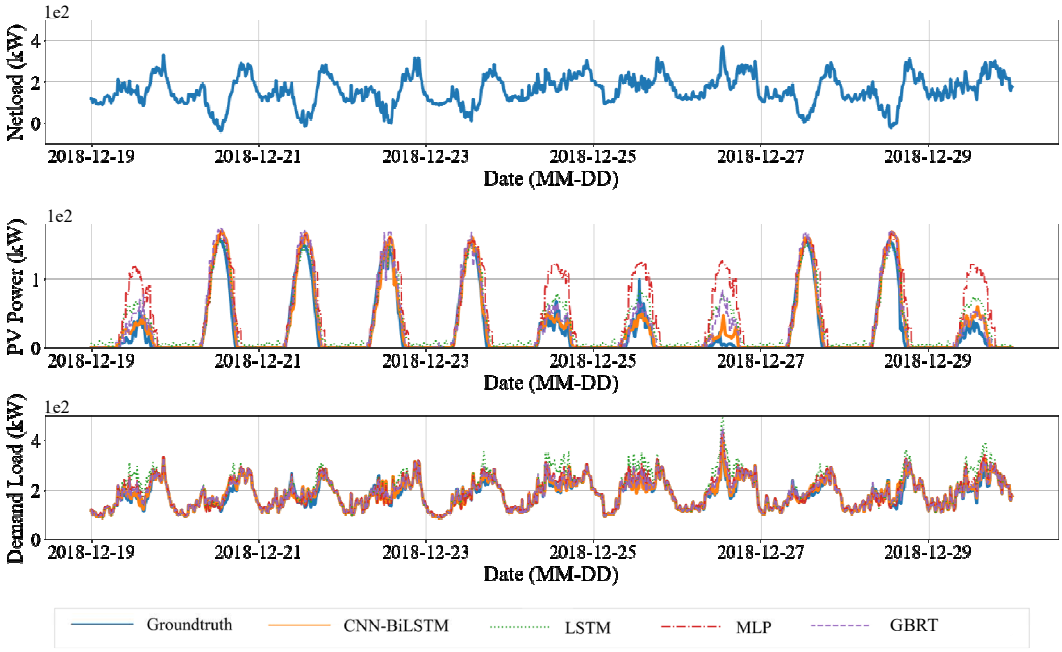


Figure 7-10. Decoupling performance for the feeder with $\bar{P}_L$ =948 kW and α=20%.

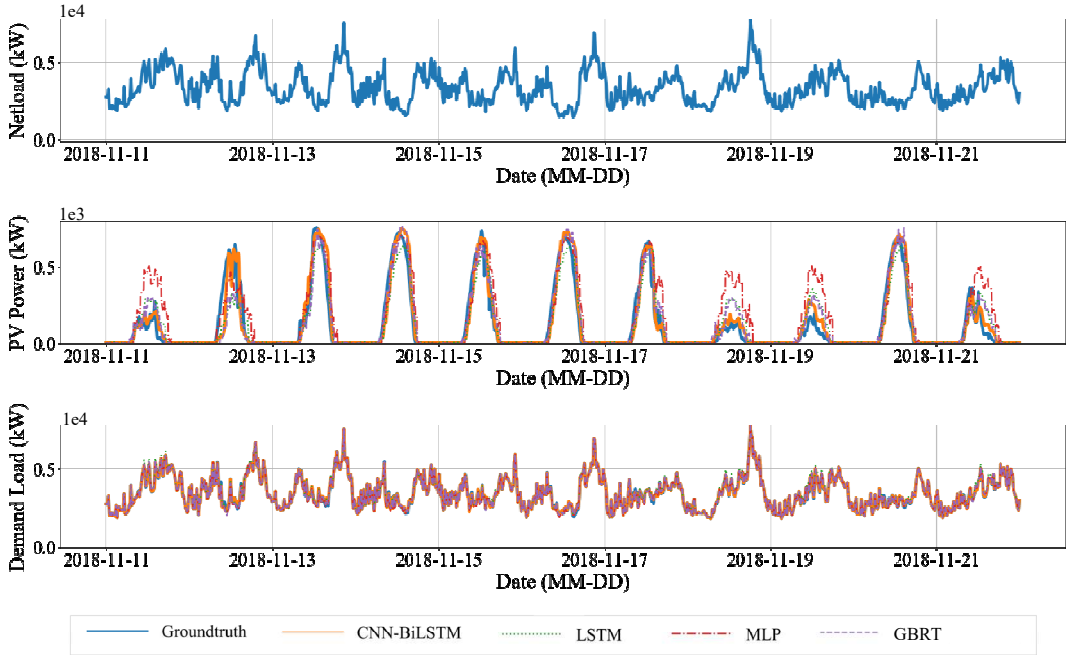
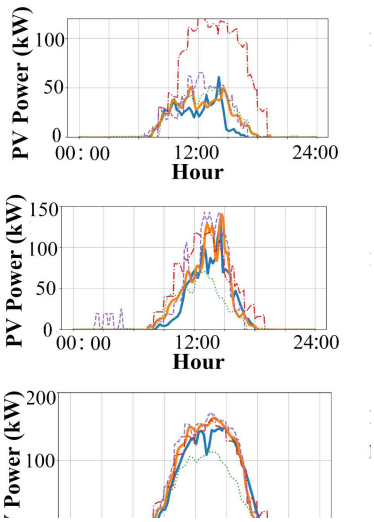


Figure 7-11. Decoupling performance for the feeder with $\bar{P}_L$ =17021 kW and α=5%.

Several estimation examples under different weather conditions are presented in Figure 7-12. Three weather conditions: clear sky, overcast, and rain. Three typical days are given for each weather condition category. The figure shows that all four algorithms track the ground truth curve very well on clear sky days. While on rainy and overcast days, since the sun is covered intermittently, these algorithms cannot always catch up with the actual PV outputs. Poor performance is observed in the MLP algorithm, as a huge error is detected between the actual and estimated curves. When it turns to the proposed CNN-BLSTM, which is the solid orange line in Figure 7-12, it can precisely track the ground truth PV outputs even during rain and overcast days.

7.4.4 Case study III: Transductive transfer learning

The training and testing data from the previous case study is used consistently from the same dataset. However, it is difficult to obtain labelled data at the target location. A transductive transfer learning approach is proposed to overcome the limitation, which forms a major hurdle for real word industrial applications. The definition of transfer learning is defined as follows:



GBRT

Figure 7-12. Examples of the estimation results of four disaggregation algorithms under different weather conditions (sunny, rainy, cloudy).

Definition 1 (Transfer Learning) [391]**.** *Providing a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, the purpose of transfer learning is to help improve the performance of the target function $\mathcal{F}_T$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$.*

While in a transductive transfer learning task, a source learning task $\mathcal{T}_S$ and target learning task $\mathcal{T}_T$ are the same (to implement decoupling task), but the domains of source and target may be different ($\mathcal{T}_S$ is a synthetic dataset, and $\mathcal{T}_T$ is real-time data in this case) [392]. In this work, the transductive transfer learning framework can be split into four steps (see Figure 7-13):

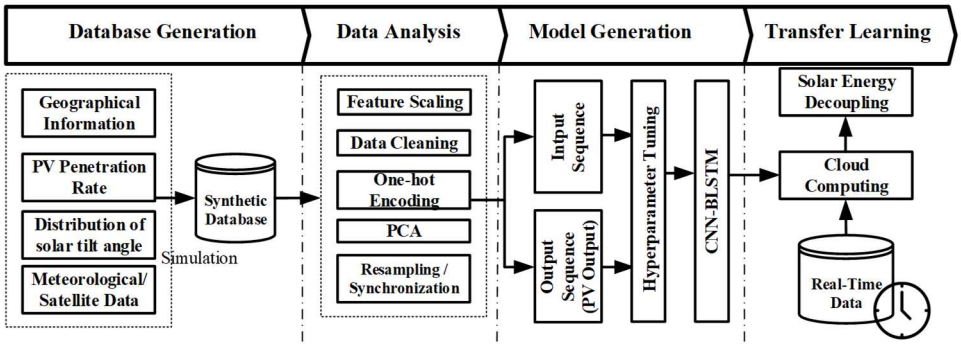


Figure 7-13. Block diagram of the transfer learning process.

Step 1 – Synthetic Database Generation: Using local conditions (such as load capacity, geographical information, the portion of PV tilt angles, meteorological data, etc.), simulation software is adopted to generate synthetic solar energy and demand load datasets. Hence, the dataset is labelled and can be used for supervised learning.

Step 2 – Data Analysis*:* As introduced in Section 7.2, the generated dataset is pre-processed to provide normalized, featured extracted data.

Step 3 – Model Generation*:* With processed data, the CNN-BLSTM neural network model is trained, and the model parameters are stored in a cloud server.

Step 4 – Transfer Learning: The unlabelled real-time data from the target area is then sent to the trained model, while the cloud server decouples the net load into solar energy and demand load. In the simulation, two cases are studied to investigate the proposed transfer learning method in Austin, Texas, and New York, detailed description of the cases is shown in Table 7-7. SAM simulation software [384] is adopted to generate a synthetic residential solar energy dataset referring to relevant information. The data provided by Dataport is adopted as real-time measurements.

Table 7-7. Relevant information about the target area.

Case	Location	Year	Load Capacity(kW)	Penetration Rate (%)	Optimal Tilt Angle(°)
1	Austin	2018	948	20	28
2	New York	2019	17021	5	34

The performance of the disaggregation system in transfer learning is presented in Table 7-8 and Figure 7-14. The CNN-BLSTM model is pre-trained via the synthetic dataset and then applied to the aggregated real-measured demand load. In Case 1, where the target area is selected in Austin, Texas, the transfer learning almost researches the equal performance in Case study 1; as for Case 2, where the target area is selected in New York, the performance is slightly worse than the model which is trained via real-measured dataset. This is because there is a minor geographical information error between the location of the synthetic data and real-measured sites. The two cases show that the proposed transfer learning method is easy to implement anywhere else, and a desirable accuracy can be achieved.

However, there are limitations of the proposed solar energy separation method: Firstly, the penetration rate of rooftop PV is extremely low in many developing countries such as India, China, and Africa, so the proposed method is only suitable for the countries with deep PV penetration rate.

Table 7-8. The performance of the disaggregation system in transfer learning.

CASES	R^2	MAE (kW)	nRMSE2 (%)	RMSE (kW)
1	0.96	5.03	5.14	9.74
2	0.79	72.19	3.70	125.94

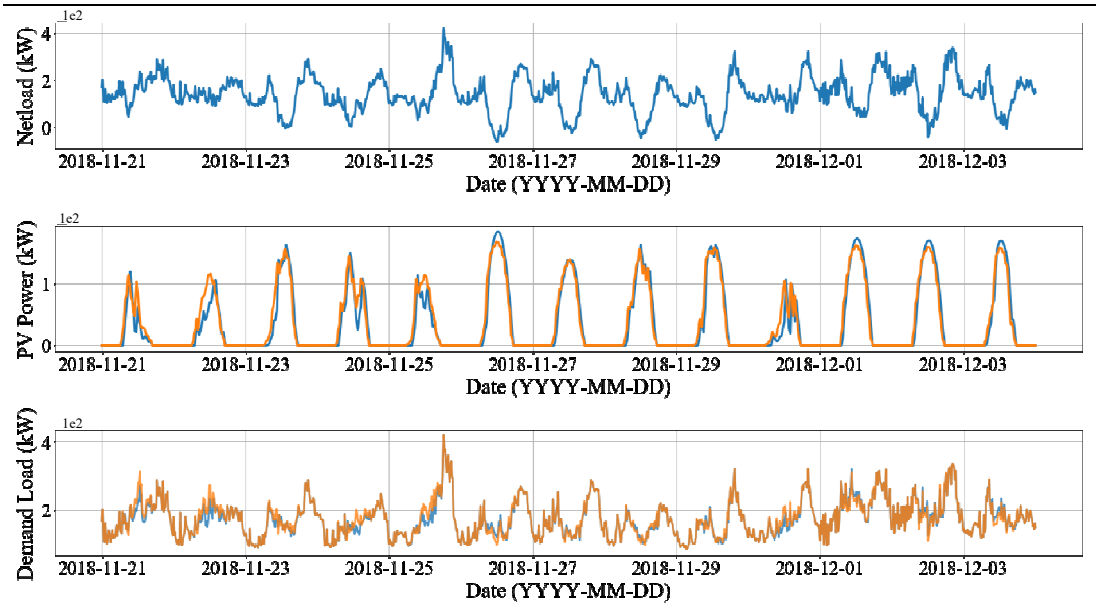


Figure 7-14. The performance of the transfer learning in Austin, Texas, US.

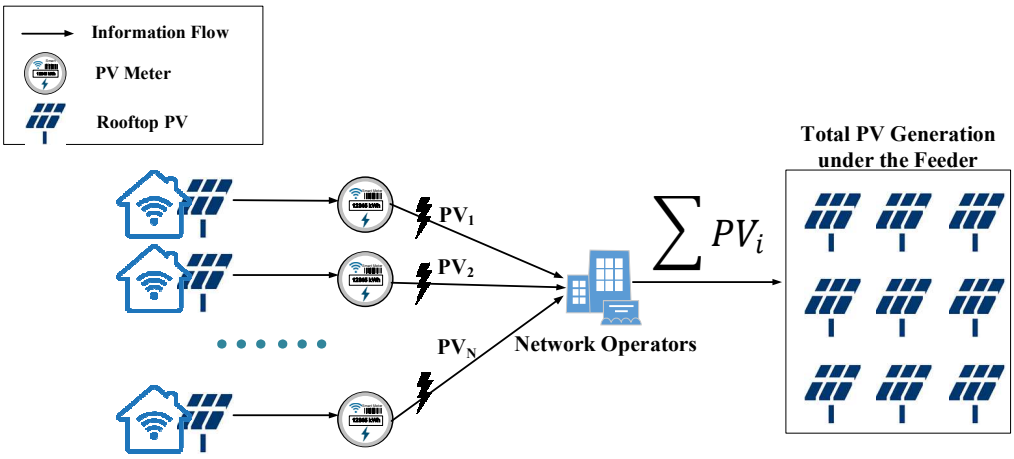
7.5 Privacy Analysis

In this section, the privacy risk of the proposed solar energy decoupling system is analysed by comparing it with the existing method. As shown in Figure 7-15, in the existing smart metering system, the DNO need to access the PV meters inside the houses to estimate the PV generation in a certain area by accessing the readings from all PV meters. Such an approach not only introduces privacy risks but also increases the cost. Detailed risks and disadvantages can be summarized as follows:

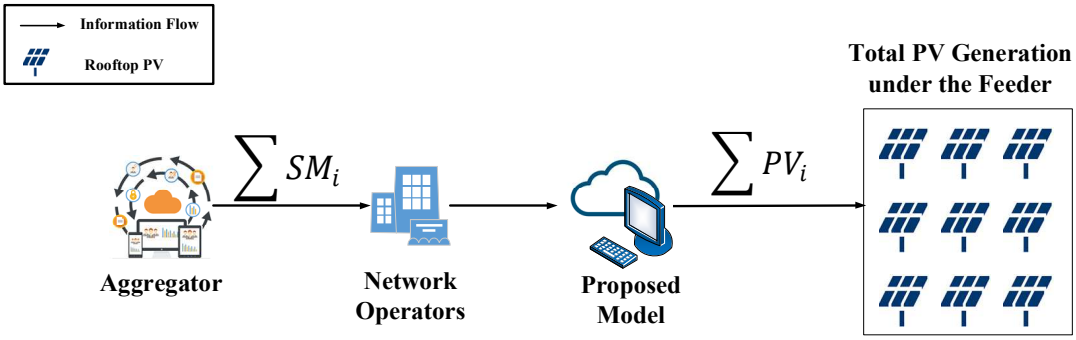
1) The direct access to individual consumption/generation data for grid operation purposes contradicts the data access policy of OFGEM [41] and BEIS [12].

Referring to the data access policy, the individual data should be aggregated or anonymized before transmitting to the DNO.

2) By sharing the PV generation information with the DNO, external adversaries can obtain the information that may eavesdrop on the communication between the PV meter and the DNO.



(a) Information flow of the Existing PV generation decoupling scheme.



(b) Information flow of the proposed PV generation decoupling scheme.

Figure 7-15. Information flow of existing/proposed PV generation decoupling scheme.

In turns of the proposed solar energy decoupling scheme, only the aggregated smart meter provided by the aggregator, designed in Chapter 4, is shared with the DNO, and the DNO has no authority to access the individual meter readings, and no extra PV meter is required. In such case:

1) Given the aggregated data and the external information, the proposed model decouples the overall PV generation from the aggregated demand. The proposed scheme strictly follows the data access regulation made by OFGEM [41] and BEIS [12]; both the input and output of the proposed model are aggregated data without accessing any individual information.
2) The external adversary does not have the opportunity to obtain the individual consumption/generation data as such information has never been shared.

Based on the analysis above, the privacy risk in the current PV energy decoupling scheme is reduced.

7.6 Chapter Summary

This chapter develops a deep learning-based solar energy disaggregation system to decouple the solar energy generated by rooftop PV systems and the real demand load from the net load measured by a feeder-level smart meter. The system collects various information from different resources, including AMI data, meteorological data, satellite-driven irradiance, and temporal information. A 1D CNN bidirectional LSTM algorithm is developed to estimate the solar energy generated in the target area. Compared to the benchmark algorithms (model-based method, upscaling model, machine learning-based models and deep learning-based methods), the precision and effectiveness of the proposed method are verified via several case studies. The influence of the PV penetration rate and feeder load capacity on the proposed system are fully investigated. The results show that the proposed method can decouple solar energy with a low error, even at a low penetration rate (5%). Moreover, the method is robust since it can be adapted to different feeder models, and the model can be trained via a synthetic dataset and still achieves desirable performance in real-world measurement. These characteristics enable the proposed system to be widely adopted and implement practically.

Chapter 8 Multi-Quantile Recurrent Neural Network for Distribution-Level Probabilistic Energy Disaggregation with Aggregated Smart Meter Data

8.1 Introduction

8.1.1 Motivation

Demand Response (DR) plays a critical role in the future smart grid, and it has flexibility in controlling and managing end-use consumers' power consumption patterns. As a result, peak demands are reduced, and the mismatch between generation and demand is minimized. In incentive-based DR schemes, the utility would control specific controllable loads directly during a certain period for load shaping. So, understanding the portion of controllable loads is vital for the utility to design DR strategies.

Loads can be divided into critical loads and controllable loads [393]. Meanwhile, controllable loads can be divided into Thermostatically Controlled Loads (TCLs) and non-thermostatically controlled loads (non-TCLs). TCLs (e.g., Heating, Ventilation And Air Conditioning (HVAC), Air Conditioner (AC), heat pumps, furnaces, and refrigerators) occupy 30-40% of the overall demand load [394], and TCLs are widely adopted in DR for their thermal inertia capability. Recent research shows that by optimizing the operation of HVAC systems, 45% of energy would be saved [395]. Moreover, EVs have high flexibility in scheduling the charging/discharging slot, benefiting DR by shaving the peak load. In addition, the high penetration of renewable

energy (e.g., solar energy) masks the ground truth demand load. However, most DR frameworks are planned for pure load demand, the invisibility of the actual load demand raised by the renewable energy influences the efficiency of the existing DR schemes. Hence, it is essential to increase the visibility of the load components by disaggregating the net load measured at feeder/substation into renewable energy generation, TCLs, and non-TCLs.

Accurate models of power system loads are vital for the simulation and prediction of the dynamic status of electric power systems. Having accurate models of the loads that can reliably reflect the underlying phenomena of the physical loads is important for designing automatic control systems and optimising their configuration. More importantly, the dynamic properties of power system loads significantly impact system stability.

The purposes of implementing feeder-level energy disaggregation are listed as follows:

(1) Real-time substation-level energy disaggregation can help the power system operators and demand-side managers to improve the system reliability, economic efficiency, and environmental impact.
(2) Help the utility better understand the overall real-time performance of the power system.
(3) Deal with the stability problems raised by the integration of renewable energy.
(4) Better understand the voltage distortion problems caused by the nonlinear loads, such as lighting, motors, electronic devices, etc.
(5) Reduce unnecessary investment in smart meters to be installed behind the individual solar panel and reduce the privacy issues raised by the smart meter.

Moreover, A system operator can estimate the real-time balancing reserve requirement by estimating the production of distributed generation resources. A utility can better plan demand response actions by knowing the weather forecast and the real-time portion of weather-dependent loads (e.g., air conditioners, heaters, dehumidifiers). A demand response provider can (1) Optimize capacity bids into ancillary services

markets using its estimate of the real-time, aggregate, demand-responsive load. (2) Use its estimate of the real-time, aggregate, demand-responsive loads as a feedback signal in load coordination algorithms [205].

8.1.2 Knowledge gaps in the existing work

The knowledge gaps in the existing work can be summarized as follow:

1) Existing feeder-level energy disaggregation requires the DNO analysis to access an individual's meter and use the NILM data mining algorithm to infer detailed appliance information, which introduces privacy risks to the energy consumers.
2) The correlation between the feeder-level demand load and external variables (weather variables, temporal variables) is not investigated thoroughly.
3) The Traditional DNN model can only make point prediction, while the Prediction Intervals made by the probabilistic model has much practical significance.

8.1.3 Chapter contribution

With the benefit of the proposed smart metering system, the regional aggregated data is available to the DNO. Hence, the chapter will investigate how to obtain load components under the feeder with only aggregated data. In this chapter, a feeder-level probabilistic energy disaggregation scheme is proposed. Detailed novelties of this work are listed as follows:

(1) The scheme utilizes a Multi-Quantile Long Short-Term Memory Neural Network (MQ-LSTM) to disaggregate various components (TCLs, Non-TCLs, PV generations, and other loads).
(2) A transfer learning model is introduced to transfer the energy disaggregation model trained with a public dataset to a local dataset. The transferability solves the issue of the data shortage.

8.1.4 Chapter structure

The rest of the chapter is organized as follows: The preliminaries, including the problem statement, and data preparation, are introduced in Section 8.2. The energy disaggregation methodology is demonstrated in Section 8.3. In Section 8.4, three case studies, which compare the proposed load forecasting algorithm and other methods, and the transferability of the model, are implemented. The conclusion and final discussion are drawn in the last section.

8.2 The Preliminaries

8.2.1 Problem statement

The target of this chapter is to disaggregate the overall feeder-level load demand into four components, which are: TCLs, non-TCLs, renewable generation, and Other Loads (OL) in both real-time and offline mode (See Figure 8-1 (a)). Figure 8-1 (b) presents the percentages of different loads under the feeder demand, referring to [327]. Among all loads, AC load and furnace load account for around 45% of the overall use in the US, as indicated in Figure 8-1. AC and furnace load also play a vital role in the DR programmes as these loads, as these loads can resist frequent and short interferences without reducing the end-use performance significantly. Moreover, EV load increases dramatically as the global EV market size has grown to 4093 thousand units in 2021 [396]; estimating the EV load will help power system operators better understand the change in demand load patterns. Furthermore, most household PVs are BTM and cannot be detected by the electricity meters, while these BTM renewable energy generations reshape the demand load shapes and cause some serious stability problems, such as the California electricity crisis [397]. Hence, separating the PV generations will increase the visibility and forecasting of the power system. Based on the analysis above, AC, furnace, EV loads, and PV generation are selected as the cases to be studied in this work, while AC and furnace loads are TCLs, and EVs are non-TCL loads rooftop PVs are renewable energy generation. Assuming the feeder-level

net load is measured as Net_t^{feeder}, the problem can be expressed as the following formula:

$$Net_t^{feeder} = \mathrm{L_t^{feeder}} + \mathrm{G_t^{PV}} + \varepsilon_t \quad \text{(8-1(a))}$$
$$= L_t^{TCL} + L_t^{non-TCL} + L_t^{OL} + G_t^{PV} + \varepsilon_t \quad \text{(8-1(b))}$$
$$= L_t^{AC} + L_t^{FURN} + L_t^{EV} + L_t^{OL'} + G_t^{PV} + \varepsilon_t \quad \text{(8-1(c))}$$

where L_t^{feeder} is the total demand load, L_t^{TCL} is the TCLs demand, $L_t^{non-TCL}$ is the non-TCLs demand, L_t^{OL} is OL demand, G_t^{PV} is the PV generation, L_t^{AC} is the AC demand, L_t^{FURN} is the furnace demand, L_t^{EV} is the EVs demand, and ε_t is the random noise.

8.2.2 Comparison among similar problems

Three similar problems in this chapter should be distinct: feeder-level energy disaggregation, load forecasting and house-level NILM. Household-level NILM is a technique for obtaining individuals' appliance consumption from overall household-level power consumption without installing intrusive sensors, such as a smart plug or smart sensors. Since most appliances have unique load curve or voltage curve characteristics, it is easy to separate every appliance from the overall load with algorithms such as HMM, RNN, and KNN. Nevertheless, when the situation comes to feeder level or distribution level, the load curve is highly aggregated and contains 200 to 4000 houses, referring to the standard feeder models provided by GridLAB-D [326]; the characteristic of the single appliance is difficult to be detected with power measurements only. Meanwhile, load forecasting technology aims to predict demand load with both long-term and short-term horizons, given historical demand load data.

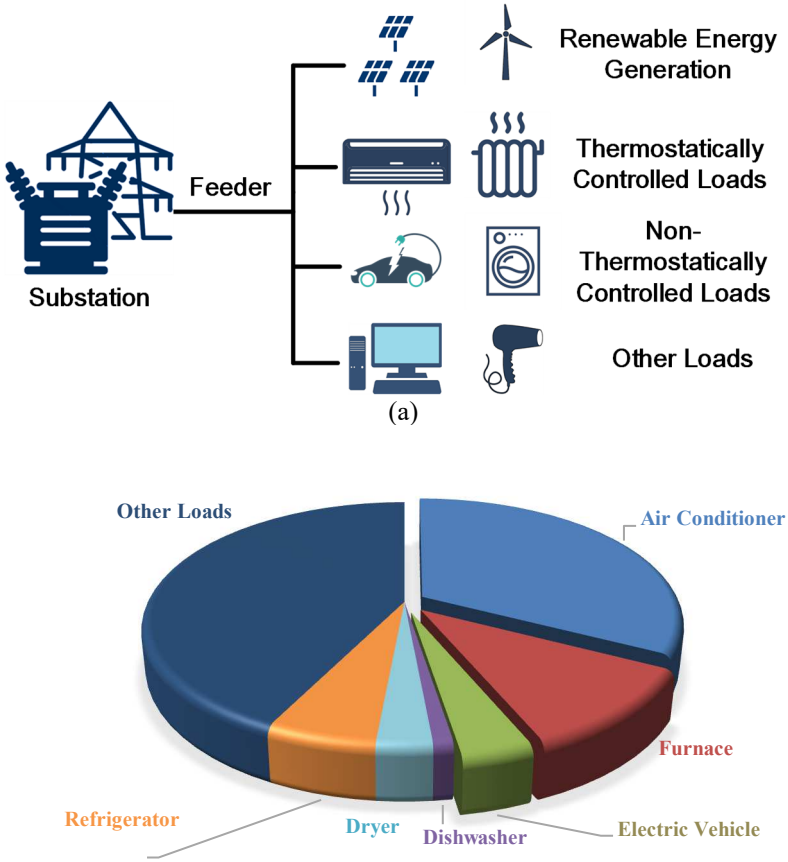


Figure 8-1. (a) Load components under substation/feeder; (b) Portion of loads under the feeder demand (Data source: Pecan Street Dataport [327]).

8.2.3 Input variables and data analysis

Input variables of the energy disaggregation system are classified into four categories: feeder, meteorological, time, and solar irradiance, as shown in Table 8-1. Meanwhile, all input variables can be divided into two categories, which are numerical variables and categorical variables. Numerical variables represent the values that can be measured and placed logically. By contrast, categorical variables take values that are names or tags, and the number of potential values is often limited to a fixed series. These categorical variables cannot be recognized by DNN models and must be converted into a numerical form. The conversion method adopted in this work is one-hot encoding. Instead of providing a single integer only, one-hot encoding provides a

set of binary variables. A detailed description of the input variables is presented as follows:

8.2.3.1 Feeder-level Demand and Appliance Load Data

The feeder models used for this research are selected from standard feeder models provided by GridLAB-D [326]. To construct the feeder-level demand load data, individual household-level smart meter data from Dataport [117] are added to match the feeder model's capacity. Dataport is the world's largest residential energy dataset, and it contains more than 700 houses; and each house measures around 20 electrical appliances in Texas, US. Hence, Dataport not only measures household electricity consumption but also provides detailed appliance usage data. Such dataset enbaMoreover, the interval resolution of the smart meter data is 15 minutes. Household-level smart meter data from Dataport was randomly drawn with replacement and added together until the total residential signal's mean R2-25.00-1 feeder model, resulting in 3 691 total houses aggregated data reaching 17021 kW. To construct the dataset for load components (AC, furnace, and EV) at the same feeder level, the demand for each household's AC, furnace and EV is summed up, respectively. The furnace, referred to as a heater or boiler in British English, is a major component of a central heating system. Normally, the fuel source of the furnace is natural gas; the furnace heats air and distributes heat to the entire building [398]. It is noticed that the furnace in the Texas area not only plays the role of a heating system to provide heat during winter but is also used to circulate cooled air during other seasons. The data is split into a training dataset (1st January 2018 to 1st August 2018), a validation dataset (2nd August 2018 to 15th September), and a testing dataset (16th September to 31st December 2018), respectively. The aggregated demand load is the model's input, while the aggregated appliance load is the output of the deep learning model.

Table 8-1. Input variables of the energy disaggregation model.

Feature type	Description	Mark
Feeder measurement		
Feeder active power flow	One week (672) lagging values and current values	$L_{t-672}, L_{t-671}, \cdots L_{t-1}, L_t$
Meteorological measurement		
Past temperature values	One week (672) lagging values and current values	$T_{t-672}, T_{t-671}, \cdots T_{t-1}, T_t$
Humidity	Humidity in current time step	HM
Wind speed	Wind speed in current time step	WS
Pressure	Pressure in current time step	P
Weather description	10 binary values for each weather condition	$WD_1, WD_2 \cdots WD_{10}$
Cloud cover	Cloud cover rate in current time step	C
Calendar information		
Day type	2 binary values for each type of day (weekday/weekend)	D_1, D_2
Holiday	4 binary values for a normal day or current/previous/after day is a holiday	L_1, L_2, L_3, L_4
Season	4 binary values for each season in one year	S_1, S_2, S_3, S_4
Month	12 binary values for each month in one year	$M_1, M_2 \cdots M_{12}$
Hour	24 binary values for each hour in one day	$H_1, H_2, \cdots H_{24}$
Solar irradiance for PV separation		
GHI	GHI in current time step	GHI
DNI	DNI in current time step	DNI
DHI	DHI in current time step	DHI
Latitude	Latitude of the PV site	Lat
Longitude	Longitude of the PV site	$Long$

8.2.3.2 Meteorological Measurement

The demand load and PV generation strongly correlate with meteorological data, so it is vital to include meteorological measurements into the input variables. In this chapter, meteorological data resources from the geographical point N 30° 15' 59.9976", W 97° 43' 59.9880" are used; NCEI [193] provides public access to the US's national historical weather data and information. Numerical variables, ambient temperature $\boldsymbol{T}$, humidity $\boldsymbol{HM}$, pressure $\boldsymbol{P}$, wind speed $\boldsymbol{WS}$, cloud cover $\boldsymbol{C}$ are chosen from the NCEI dataset. For ambient temperature, 673 variables are generated, spanning the last week and current temperature measurements:

$$\boldsymbol{T} = [\boldsymbol{T_t}, \boldsymbol{T_{t-1}}, \cdots \boldsymbol{T_{t-671}}, \boldsymbol{T_{t-672}}] \tag{8-2}$$

Figure 8-2 shows the correlation between $\boldsymbol{T}$ and different loads at the feeder level. The figure shows that TCLs are highly influenced by temperature and relevant weather variables (e.g., humidity, wind speed). AC has a positive correlation with temperature; as $\boldsymbol{T}$ increases, the power consumption of AC raises as well. Since the furnace has a dual role (heating and circulating) in this research, when $T < 13℃$, the correlation between $\boldsymbol{T}$ and furnace demand load is negative, and when $T > 18℃$, the correlation turns positive. Meanwhile, $\boldsymbol{T}$ has little influence on the non-TCL demand (such as EV) as the curve is flat throughout different temperature periods.

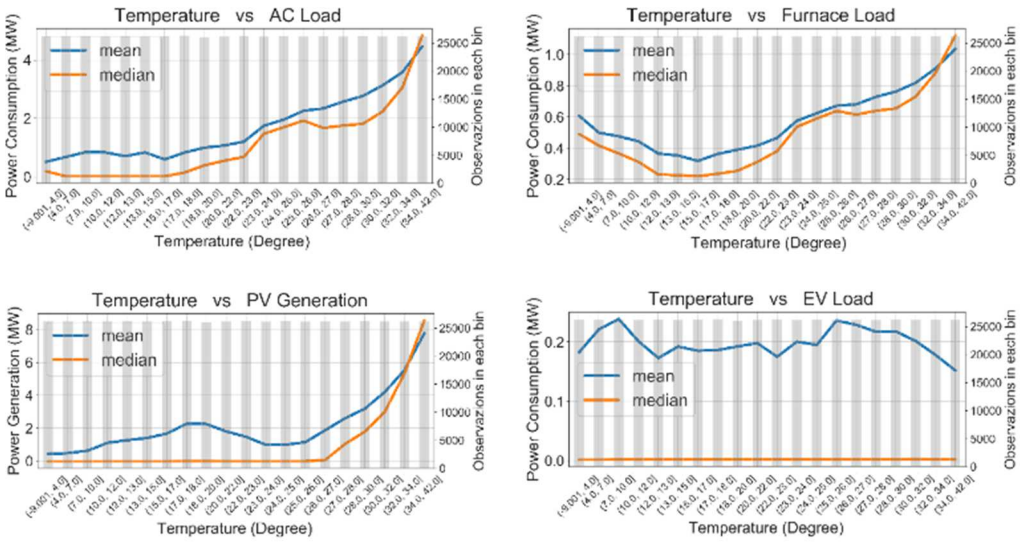


Figure 8-2 shows the correlation between temperature and different loads (Data source: Pecan Street Dataport [327]).

The heatmap in Figure 8-3 shows the Pearson Correlation (ρ) of loads and different meteorological variables (temperature, pressure, humidity, and wind speed); a higher value of ρ represents a stronger correlation between two variables. From the figure, it is observed that these meteorological variables impact AC, furnace loads, and PV generations, while they have little influence on the EV load, as the EV load is related to user's behaviour patterns rather than weather conditions. Moreover, the temperature has the strongest correlation with all loads except EV. Pressure, humidity, and wind speed also influence AC, furnace, and PV to different degrees. Finally, categorical meteorological data and weather description WD also have an extraordinary impact on the demand load. Ten different weather conditions are described in the NCEI

dataset, which are: Mist, Clouds, Snow, Clear, Rain, Drizzle, Haze, Thunderstorm, Fog, and Dust.

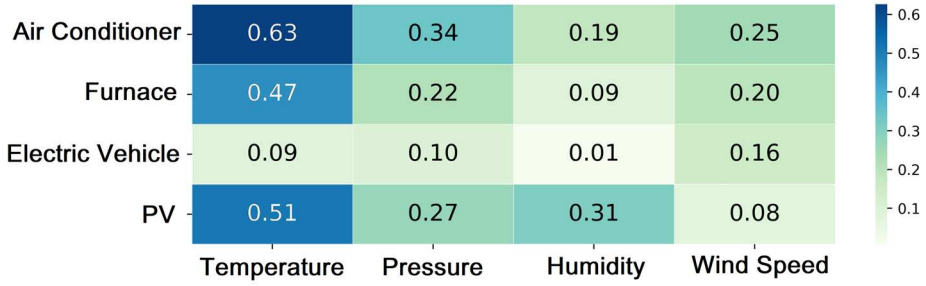


Figure 8-3. Pearson Correlation of loads and meteorological variables (Data source: Pecan Street Dataport [327]).

8.2.3.2.1 Calendar Data and Holiday Information

Calendar data and holiday information is other vital factor influencing consumers' behaviours and the electricity events that happen inside their houses. As shown in Table 8-1, time information variables include:

- **Type of the Day.** Day types include weekdays, weekends, and holidays. Including the day types enables the disaggregation model to be sensitive to the week's variation. In Austin, Texas, 14 days are marked as a holiday in 2018, referring to [399]. Considering the influence of holiday on residents would span before or after the holiday, the day before and after the holiday is also viewed as new variables. Hence, four binary variables are used to represent a holiday. Figure 8-4 compares typical load profiles during weekday, weekend, and holiday. The peak loads of overall demand load, TCLs, and EV are higher than other types, while the peak of loads during weekends is clipped. A dramatic reduction of all demands is observed during holidays, especially for furnace and AC loads. This is due to some residents leaving their houses to travel elsewhere rather than stay home.

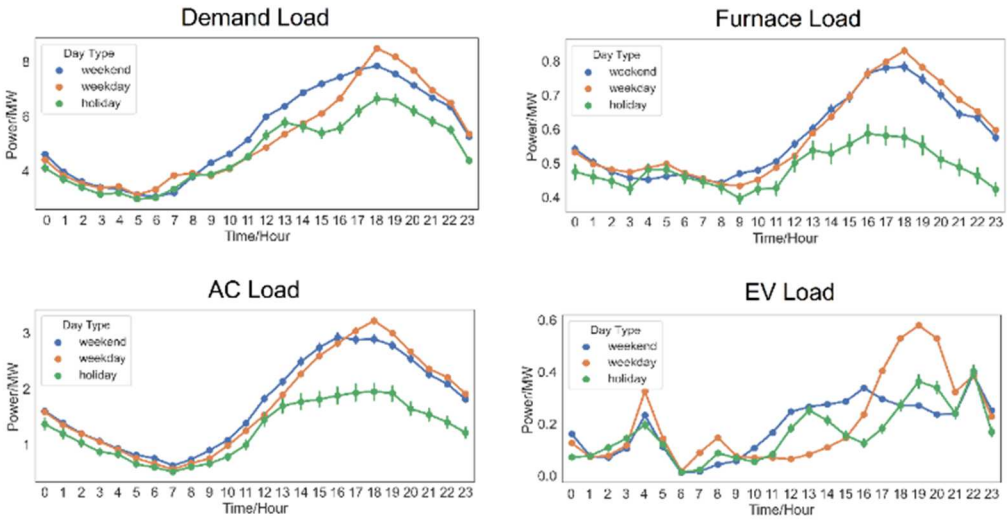


Figure 8-4. Net/appliance load profiles under different day types.

- **Season *S*.** Seasonal variation (Spring, Summer, Autumn, and Winter) is also a critical factor influencing the demands, and consumers prefer different electricity appliances during different seasons. For instance, AC is typically used during summer cooling, and the heating system is preferred in winter for heating purposes. In this work, four binary variables S_1, S_2, S_3, S_4 are used to represent the season, e.g., [0,0,0,1] represents spring. Figure 8-5 uses the Stats-Violin plot to show the distributions of the load demand of different load components in four seasons. The figure shows that the power consumption of AC load in summer and autumn is much larger than in spring or winter. As for the furnace plays a dual role as a heating system and air circulation device, the demand for the furnace is high in both summer and winter. The distribution of EVs does not show any difference among various seasons, which shows that the EV charging/discharging activities are not influenced by seasons. Finally, solar energy generation significantly influences the season as the PVs generate more power during summer and autumn.

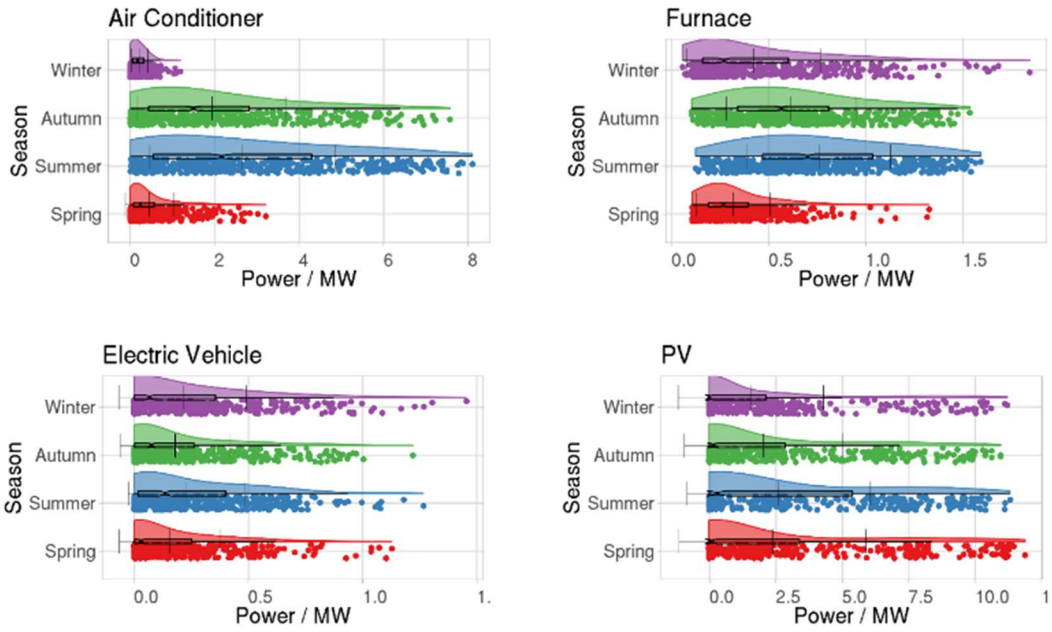


Figure 8-5. Stats-Violin plot of appliance load profiles under seasons.

- **The Hour of the day H and the month of the year M.** Twenty-four binary variables and 12 binary variables are used to represent hour and month, respectively.

8.3 Energy Disaggregation Scheme

8.3.1 System overview

The framework of the proposed multi-quantile RNN energy disaggregation system contains two modes, online mode and offline mode, shown in the block diagram of Figure 8-6. In offline mode, the energy disaggregation model is trained with a historical dataset, and offline analysis is also implemented for grid planning and arrangement purpose. The real-time net load measurement is disaggregated in online mode into individual load components.

8.3.1.1 Offline training with a public dataset

The offline mode has two functions: (1) training the DNN model and uploading the trained model parameters to the cloud server; (2) analysing the load components of historical feeder demand. Typically, the power utility or third parties such as software companies should operate the offline mode to help the utility build the energy disaggregation models.

Step 1: Historical data loading. Historical data is loaded from the historical database. The database contains historical meteorological measurements, historical calendar/holiday data, historical solar irradiance data, and historical smart meter data. The historical smart meter data contains both household-level and appliance-level power consumption data, so the smart meter data can be aggregated to generate feeder-level demand load. Since the historical appliance-level smart meter data is not always available for the energy utility, there are several approaches to obtain such label training data: (1) Utilize the public dataset at the research location, such as Dataport [117] for the distribution network in Texas, US. Such a public dataset is anonymized and under the permission of consumers. (2) Utilize software simulation software to generate synthetic netload and detailed appliance-level smart meter data. (3) Utilize the transfer learning method as introduced in Section 8.5.3. (4) Select a small group of volunteers under the distribution network, and the energy utility collects the smart meter data and the appliance usage information inside the volunteers' houses under their permission. Since the trials are already under the volunteers' consent, such trials will not cause privacy issues.

Step 2:Training the PV separation model. The PV separation model is trained via the process introduced in Chapter 7.

Step 3: Training load disaggregation model. Like Step 2 mentioned above, Step 3 trains the offline model to implement demand load disaggregation. However, there are two different points between the two steps. Firstly, the input variables and outputs of

the models are different. Compared to the PV separation model, the load disaggregation model does not require solar irradiance data.

In contrast, holiday information is added to the inputs as such special events highly influence manual activities. Secondly, instead of taking PV generation as output, the load disaggregation model takes the portions of each load component as outputs. Furthermore, the trained model and model parameters are uploaded to the cloud server for online estimation purposes.

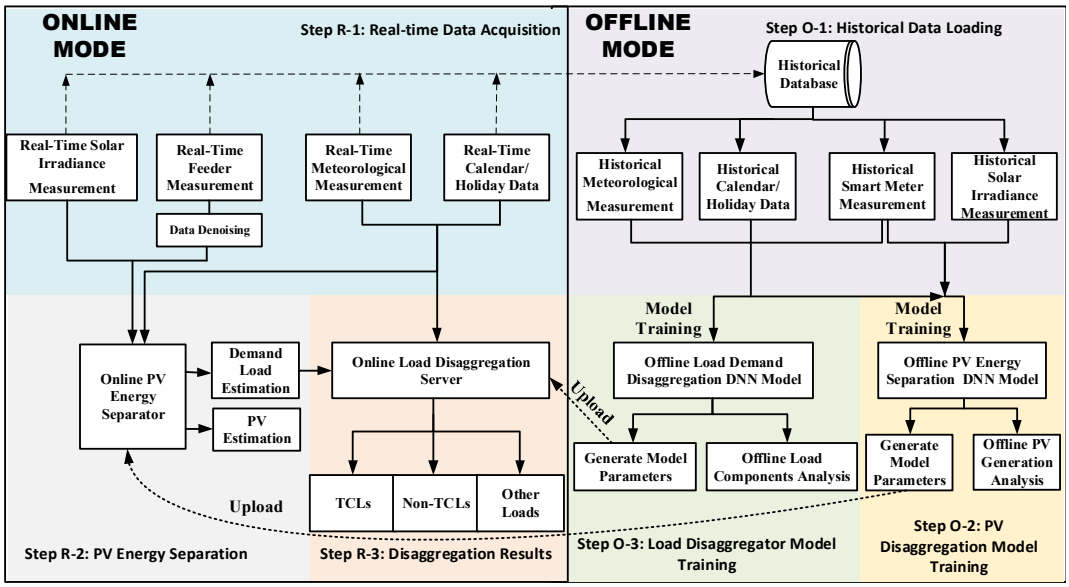


Figure 8-6. Online/Offline PV energy disaggregation framework.

8.3.1.2 Online Mode

In online mode, the power utility would like to use the online server to analyse the load components on a real-time basis. The models trained in offline mode are uploaded to the online server, so the utility can implement online computing without training the models simultaneously.

Step 1: Real-time measurements collection. The utility receives the real-time feeder demand measurement from the feeder-level smart meter or the DSCADA system. Meanwhile, the utility can also access the real-time meteorological measurement the local weather station provides. The real-time calendar data are generated by the system or online server such as Google Calendar [400]. Moreover, holiday data are provided

by the local government. All real-time measurements are synchronized and pre-processed (normalization for numerical variables and one-hot encoding for categorical variables) before feeding into the online server. In addition, the noise in the feeder measurement and communication would influence the performance of the disaggregation server. Hence, a data denoising method is adopted to filter out noise. A detailed description of the denoising method is introduced in the following section. Primarily, real-time satellite solar irradiance data provided by NCDC [21] is also obtained to separate solar energy components. All real-time data is also saved into a historical database to frequently update offline models.

Step 2: Real-time PV generation separation. Before disaggregating the feeder demand into individual load components, the PV generation components are separated from the net load as the negative loads would impact the detection of other positive loads. The online PV separating process is introduced in 7.3.3.

Step 3: Real-time demand load disaggregation. The estimated demand load of the PV separator is then fed into an online energy disaggregation server along with real-time meteorological measurements and calendar/holiday information. Obtaining the DNN model and model parameters from the offline mode, the online load disaggregation server then disaggregates the estimated demand load into three components: TCLs, Non-TCLs, and OLs.

8.3.2 Domestic loads disaggregation at feeder-level

The demand load L_t^{feeder} estimated by the solar energy separator introduced in Chapter 7 is then used as input variables of the energy disaggregation model. The purpose of the model is to separate the demand load L_t^{feeder} into TCLs (AC and furnace), Non-TCLs (EV) and OL, as illustrated in :

$$L_t^{feeder} = L_t^{TCL} + L_t^{non-TCL} + L_t^{OL} \tag{8-3}$$

In this work, one-week historical load demand $\boldsymbol{L}_F$ with interval 15 min is generated as the input variables of the energy disaggregation model:

$$\boldsymbol{L}_F = \left[L_t^{feeder}, L_{t-1}^{feeder}, \cdots, L_{t-671}^{feeder}, L_{t-672}^{feeder}\right] \tag{8-4}$$

The core component of the energy disaggregation model is the multi-quantile long short-term memory (MQ-LSTM). The detailed description of the MQ-LSTM algorithm is introduced as follows.

8.3.2.1.1 Multi-Quantile Long Short-Term Memory (MQ-LSTM)

MQ-LSTM is a technology built on traditional LSTM, and it enables the LSTM neural network to make probabilistic predictions by combining Quantile Regression (QR) with LSTM units. A more comprehensive analysis of dependent variables can be obtained by QR's measures of central tendency and statistical dispersion [401]. To implement probabilistic estimation, a set of quantiles should be set in ahead $\boldsymbol{\tau} = \tau_1, \tau_2, \cdots, \tau_M$, and M is the total quantiles number. The τth quantile (τ-quantile) of a random variable Y can be defined as:

$$q_Y(\tau) = F_Y^{-1}(\tau) = inf\{y: F_Y(y) \geq \tau\} \quad 0 < \tau < 1 \tag{8-5}$$

where $F_Y(y)$ is the cumulative distribution function of Y and can be expressed as:

$$F_Y(y) = P(Y \leq y) \tag{8-6}$$

The pinball loss function of QR is presented in (8-10):

$$\rho_\tau(\mu) = \begin{cases} \tau\mu & if\ \mu \geq 0 \\ (\tau - 1)\mu & if\ \mu < 0 \end{cases} \tag{8-7}$$

MQ-LSTM requires training M models individually, and each model is an LSTM model. LSTM is a recurrent neural network; it can process entire data sequences and learn long-term dependencies. The LSTM unit regulates information by relying on a structure known as a gate. The gate consists of a sigmoid activation function σ and a pointwise multiplication operation. The sigmoid activation function only has two

values, namely "0" and "1"; a value of “0” means the gate is closed, and “1” means the gate is open, and all information can go through the gate. There are three gates in the LSTM unit, which are forget gate $f_{t,\tau}$, input gate $i_{t,\tau}$ and output gate $o_{t,\tau}$. With the regulations of the gates, the information of the cell state $C_{t,\tau}$ is updated to retain critical information from the previous sequence.

The responsibility of the forget gate is to delete the information from the cell state C_t. As shown in (8-8), the forget gate $f_{t,\tau}$ takes two inputs, x_t and $h_{t-1,\tau}$, where h_{t-1} is the hidden state from the previous cell, and x_t is the input to the present stage. If the output of $f_{t,\tau}$ is closer to “1”, that is, to keep, or the information is forgotten.

$$f_{t,\tau} = \sigma\left(W_{f,\tau}\left[x_t, h_{t-1,\tau}\right] + b_{f,\tau}\right) \tag{8-8}$$

As for the input gate $i_{t,\tau}$, x_t and $h_{t-1,\tau}$ is passed through a sigmoid function to determine the values to be updated, see (8-9). Also, x_t and $h_{t-1,\tau}$ is passed into a tanh function to squish values between [-1,1] to create a new candidate cell state value $\widetilde{C_{t,\tau}}$, see (8-10). Finally, the new cell state $C_{t,\tau}$ is determined given $i_{t,\tau}$ and $\widetilde{C_{t,\tau}}$, shown in Equation (8-11):

$$i_{t,\tau} = \sigma\left(W_{i,\tau}\left[x_t, h_{t-1,\tau}\right] + b_{i,\tau}\right) \tag{8-9}$$

$$\widetilde{C_{t,\tau}} = tanh\left(W_{c,\tau}\left[x_t, h_{t-1,\tau}\right] + b_{c,\tau}\right) \tag{8-10}$$

$$C_{t,\tau} = f_{t,\tau} \odot C_{t-1,\tau} + i_{t,\tau} \odot \widetilde{C_{t,\tau}} \tag{8-11}$$

Finally, the output of the cell and the hidden state is determined by the output gate $o_{t,\tau}$:

$$o_{t,\tau} = \sigma\left(W_{o,\tau}\left[x_t, h_{t-1,\tau}\right] + b_{o,\tau}\right) \tag{8-12}$$

$$h_{t,\tau} = o_{t,\tau} \odot tanh\left(C_{t,\tau}\right) \tag{8-13}$$

where $W_{f,\tau}$, $W_{i,\tau}$, $W_{c,\tau}$, $W_{o,\tau}$ are weight matrices; and $b_{f,\tau}$, $b_{i,\tau}$, $b_{c,\tau}$, $b_{o,\tau}$ are the bias. Typically, one fully connected layer $z_{t,\tau}$ is connected between the LSTM layer and

output layer. The function of the fully connected layer is to convert $h_{t,q}$ into proper output size, see (8-14):

$$z_{t,\tau} = \sigma\left(W_{h,\tau} \cdot h_{t,\tau} + b_{h,\tau}\right) \tag{8-14}$$

where $W_{h,\tau}, b_{h,\tau}$ are the weight and bias of the fully connected layer, respectively. Finally,M quantiles estimations $F_\tau(x_t) = \left\{f_{\tau_1}(x_t), f_{\tau_2}(x_t), \cdots, f_{\tau_M}(x_t)\right\}$ are evaluated by the MQ-LSTM. Hence, the output of the MQ-LSTM neural network is:

$$f_\tau(x_t) = W_{z,\tau} \cdot z_{t,\tau} + b_{z,\tau} \tag{8-15}$$

and $W_{z,\tau}, b_{z,\tau}$ are the partial parameters of the τ-th quantile output layer. The MQ-LSTM model is optimized by minimising the quantile optimization function E_τ:

$$E_\tau = \frac{1}{N}\sum_{t=1}^{N} \rho_\tau\left(y_t - f_\tau(x_t)\right) \tag{8-16}$$

where N is the total number of data, y_t is the ith ground truth value.

8.3.2.2 Model Description

The techniques introduced in previous sections are combined to construct the energy disaggregation model. The model takes the MQ-LSTM as the core component, and the variables introduced in Table 8-1 are adopted as the model's input variables. Three input layers are designed since multiple independent variables are considered. As shown in Figure 8-7, The input layers take the demand load sequence $\boldsymbol{L}_F$, Temperature sequence $\boldsymbol{T}$, and other variables as input, respectively. LSTM layers are then applied to the first two input layers to process the sequence data, and a one-hot encoder is adopted to transfer categorical variables (e.g., the hour of the day, the season of the year) into numerical data. Then a concatenate is used to merge three input layers. Two fully connected layers enable the network to extract features better and learn the input data. A quantile layer evaluates multiple outputs for different quantiles.

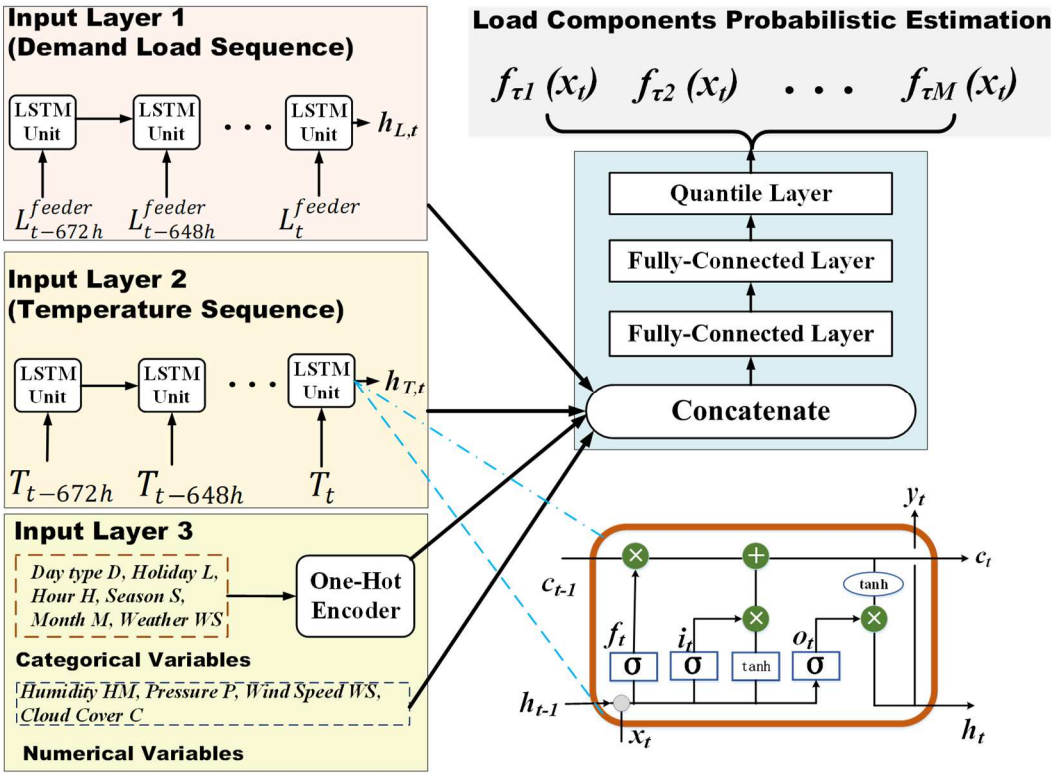


Figure 8-7. The main structure of the MQ-LSTM-based energy disaggregation algorithm.

8.4 Evaluation Criteria

8.4.1 Software and hardware platform

Various open access packages and libraries based on Python 3.7 are adopted to implement the proposed simulation case study. PyWavelets package [379] is adopted to implement the DWT-based data denoising method. Scikit-Learn package [402], and LightGBM package [258] are used for implementing Q-GBRT and Q-LGB algorithms, respectively. Moreover, TensorFlow 2 [403] is used to construct a quantile deep neural network. As for hardware, the simulation and computation are implemented on a high computation ability computer equipped with a Core i7-7700HQ CPU, NVIDIA GTX 1060 GPU 2.80 GHz (8 cores), and 8 GB RAM.

8.4.2 Performance metrics

In this chapter, three probability density prediction metrics are adopted to assess the efficiency of the energy disaggregation model.

The conventional point/deterministic prediction estimates a deterministic curve that minimises the actual curve's error. In comparison, the probabilistic prediction describes the variation of the load by providing outputs in the form of Probability Density Function (PDF), confidential intervals, or quantiles of the distribution. The uncertainty information predicted by the probabilistic model cannot be evaluated with point prediction metrics such as RMSE, MAE, and MAPE. Referring to [404], the main properties of a probabilistic forecast are summarised into four aspects: accuracy, bias, reliability (also called calibration), and sharpness. As for accuracy and bias properties, these two properties are important for both point, and probabilistic prediction. The MAE and the RMSE are widely used to measure accuracy, while the Pearson correlation coefficient (ρ) is used to measure bias. In terms of reliability and sharpness, which are two special properties of the probabilistic method, reliability quantifies the similarity between the a priori forecast probability and the a posteriori observed frequency; a forecast is reliable if it appears to be drawn from the same distribution as the observations. Sharpness is defined as the ability of a forecast to concentrate probabilistic information about future outcomes. Therefore, the metrics introduced in this thesis (Prediction Interval Coverage Probability (PICP) and Absolute Average Coverage Error (AACE), Winkler Score (WS)) are used to evaluate the reliability and sharpness, respectively. Furthermore, a metric should consider all four properties of the probabilistic prediction to make a comprehensive evaluation of the model performance, so a Continuous Ranked Probability Score (CRPS) is employed to show the overall performance.

8.4.2.1 Reliability

Reliability indicates whether the quantile regression model can efficiently capture the targets into their predicted Prediction Intervals (PIs). PICP and AACE are introduced to assess the model's reliability.

(1) PICP: As an essential metric adopted to assess probability density prediction, PICP indicates the probability that ground truth values are within the prediction interval (between lower and upper boundary). The values of PICP range from 0% to 100%, and the more significant PICP value represents more ground truth values that fall into the predicted interval. The formula to calculate PICP is:

$$PICP = \frac{1}{N}\sum_{i=1}^{N} \varepsilon_i \qquad \varepsilon_i = \begin{cases} 1, & y_i \in [L_i, U_i] \\ 0, & y_i \notin [L_i, U_i] \end{cases} \tag{8-17}$$

where N is the number of testing data, ε_i is the Boolean value, L_i is the lower boundary, and U_i is the upper boundary.

(2) AACE: AACE indicates the deviation of PICP to PINC, the expected PICP value. The equation of AACE is:

$$AACE = |PICP - PINC| \tag{8-18}$$

where $PINC = 1 - \alpha$, and α is nominal proportions. Smaller AACE represents a more precise coverage probability provided by PIs.

8.4.2.2 Sharpness

The model's performance cannot be thoroughly investigated with reliability metrics only since a more comprehensive PI can include more target points into it and achieve a higher PICP value. However, a wide PI has poor performance in tracking the variation and fluctuation of the target curve. Hence, the sharpness of the PIs is also extremely important for probabilistic estimation. Proposed by Winkler in 1972 [405]

uses WS to assess the width of the interval with a penalty once the observation is outside the interval. WS is defined as:

$$WS = \begin{cases} \Delta_i & L_i \leq y_i \leq U_i \\ \Delta_i + 2(L_i - y_i)/\alpha & y_i < L_i \\ \Delta_i + 2(y_i - U_i)/\alpha & y_i > U_i \end{cases} \quad (8\text{-}19)$$

where Δ_t is the width of PIs at time point i and $\Delta_t = U_t - L_t$. As for WS, lower scores are associated with narrower intervals and better estimates of the PIs.

8.4.2.3 Overall Performance

Finally, the overall performance of the probabilistic model is evaluated by the CRPS. CRPS measures the difference between the predicted and observed cumulative distributions functions (CDF) by considering both reliability, sharpness and accuracy [406]. Let $F(t)$ be the predictive CDF and y be the ground truth observation. The continuous ranked probability score for single observed point y, $crps$ is given by:

$$crps(F, y) = \int_{-\infty}^{\infty} (F(t) - \mathbb{1}(t - y))^2 dy \quad (8\text{-}20)$$

where $\mathbb{1}(\cdot)$ is the Heaviside function, it takes the value of 1 when $t < y$ and equals 0 otherwise. The average score of crps among N observations, $CRPS$ is used to assess the model performance:

$$CRPS = \frac{1}{N} \sum_{i=1}^{N} crps\ (F_i, y_i) \quad (8\text{-}21)$$

It is noticed that the $CRPS$ is negatively oriented, a smaller value of CRPS represents a better prediction performance, and the predicted CDF is closer to the ground truth CDF. Moreover, the unit of $CRPS$ is the same as the observed variable, which is MW in this case.

8.5 Case Study

Two case studies are introduced in this section to evaluate the proposed energy disaggregation model. The first case study compares the proposed MQ-LSTM algorithm with other advanced quantile regression models. The second case study investigates the transferability of the proposed energy disaggregation model.

8.5.1 Benchmark models

Following state-of-the-art algorithms are adopted in the case studies:
(1) Multi-Quantile Gated Recurrent Unit (MQ-GRU) [214];
(2) Multi-Quantile Convolutional Neural Network (MQ-CNN) [407];
(3) Quantile Light Gradient Boosting Machine (Q-LGB) [258];
(4) Quantile Gradient Boosting Regression Tree (Q-GBRT) [214].
The hyperparameter space of the proposed model and the benchmark models, is presented in Table 8-2.

8.5.2 Case study I: comparison of the proposed algorithms with other methods

The interval resolution of the data adopted in this case study is 15 min, and a denoising level 2 is applied to the original data to filter out noise in the measurements. Moreover, four weeks' historical feeder load demand and temperature record are adopted as input variables to enable the LSTM neural network to extract helpful information from the past. The estimated demand load L_t^{feeder} is then fed into the energy disaggregation model to obtain the power consumption of individual appliances: AC, furnace, and EV. The proposed model is compared with MQ-GRU, MQ-CNN, Q-LGB, and Q-GBRT algorithms to thoroughly investigate the performance in four metrics: PICP, WS, Score, and training time.

Table 8-2. Hyperparameter space.

Hyperparameter	**Space**
Hyperparameters of all models	
Upper bound	0.975
Lower bound	0.025
Number of quantiles	24
Loss function	Pinball loss
Q-GBRT	
Learning rate	0.1
Number of estimators	250
Maximum depth	6
Maximum number of features	5
Minimum number of samples to split	2
Minimum number of samples for a leaf	10
Q-LGB	
Boosting type	gbdt
Maximum depth	6
Number of leaves	30
Minimum number of samples for a leaf	10
Learning rate	0.1
Objective	quantile
Number of estimators	250
Early stopping rounds	20
MQ-CNN	
Number of convolutional layers	3
Number of kernels of each convolutional layer	64
Kernel size in each convolutional layer	9
The activation function in each convolutional layer	ReLU
Pooling type after each convolutional layer	MaxPooling
Pooling size after each convolutional layer	4
Number of dense layers	3
Number of neurons in each dense layer	256, 64, 32
The dropout rate of the dense layer	0.5
Batch size	512
Optimizer	Adam
Epochs	1000
Early stopping rounds	10
Learning rate	10^{-3}
MQ-LSTM/MQ-GRU	
Number of LSTM/GRU layers	3
Number of neurons in each LSTM/GRU layer	1024, 512, 128
The activation function in LSTM/GRU layer	ReLU
Recurrent dropout rate	0.3
Number of dense layers	3
Number of neurons in each dense layer	256, 64, 32
Batch size	512
Optimizer	Adam
Learning rate	10^{-3}
The activation function in the output layer	Linear
Epochs	1000
Early stopping rounds	10

The training time of each algorithm is shown in Table 8-3; the proposed MQ-LSTM, MQ-GRU, and MQ-CNN are trained on a GPU-based Tensorflow platform, while Q-LGB, Q-GBQT models are trained on a computer without GPU used. GPUs are suitable for training deep neural network models because they can process multiple

computations simultaneously. The computer installs many cores, which allows it to better and faster compute multiple parallel processes. Although a high computation ability CPU (Core i7-7700HQ CPU) is adopted, it still takes nearly 275 min (4.6 h) to finish the evaluation of the Q-GBRT model. The training time of the Q-GBRT model is much longer than other algorithms, demonstrating that the Q-GBRT model is less practical and flexible in application.

Table 8-3. Comparison of training time (min).

Appliance\Algorithm	MQ-LSTM	MQ-GRU	MQ-CNN	Q-LGB	Q-GBRT
AC	17.67	16.65	20.12	37.69	277.12
Furnace	16.78	17.23	22.32	36.26	273.45
EV	17.23	18.67	21.87	37.72	274.90
Average	17.22	17.52	21.43	37.22	275.16

In contrast, the training time of the Q-LGB model is less than the Q-GBRT model, while it only takes 37.22 min to finish the training process on average. When it turns to GPU-based training models, the training time of the proposed MQ-LSTM is 93% less than the time of the Q-GBRT model. Furthermore, the training time of MQ-GRU and MQ-CNN are shorter than 25 min. From the result, apart from the Q-GBRT model, the training time of all other models is considerable in industrial applications.

The reliability of the probabilistic models is assessed by metrics PINC and AACE, respectively. The comparison results are shown in Figure 8-8 and Table 8-4. In Figure 8-8, the PICPs of five algorithms to three appliance loads are plotted. In an ideal situation, the value of PICP is equal to PINC, as shown in the solid red curve (1:1 line). From the figure, among all curves, the proposed MQ-LSTM is closest to the ideal curve, meaning that the MQ-LSTM model is better than other benchmarks for energy disaggregation tasks and can produce reliable PIs. The maximum AACEs of the PIs evaluated by the MQ-LSTM model, which is the metric to show the deviation between PICP and PINC, are 2.79%, 8.10%, and 10.12% for AC, Furnace, EV load, respectively. As for AC load, the other four algorithms also show merit reliability as the maximum AACEs of MQ-GRU, MQ-CNN, Q-LGB, and Q-GBRT are 5.42%, 8.30%, 18.63%, and 12.63%, respectively. However, the reliability of PIs computed by MQ-CNN and Q-LGB for Furnace load is considerably low as these algorithms

overestimate PICP, and the disaggregation result is unreliable for the power system industry. Furthermore, as for EV load, PIs evaluated by all five algorithms are well-calibrated as all five curves overlap with the ideal curve precisely. To summarize, the reliability of PIs provided by MQ-LSTM is higher than any other algorithm for different load tasks.

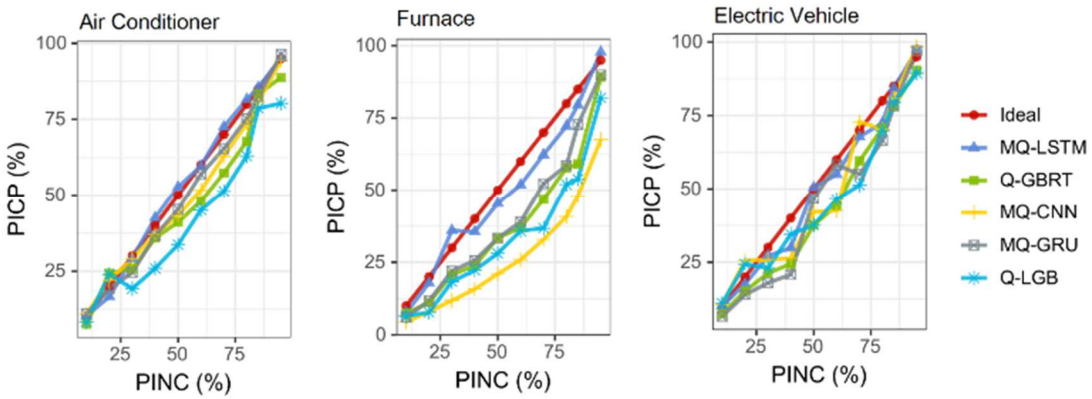


Figure 8-8. PI reliability diagrams: PICP of five algorithms as a function of PI nominal coverage.

The sharpness is assessed by metric WS, as indicated in the boxplots in Figure 8-9 and Table 8-4. From the figure, MQ-CNN is the bluntest model. The average WS of its PIs is 2.70, 3.70, and 5.10 for AC, Furnace, and EV loads. Meanwhile, the Q-LGB and Q-GBRT are the sharpest models among all algorithms, and the sharpness of MQ-LSTM and MQ-GRU is between the abovementioned models. However, it should be noticed that the sharpness should be analysed along with reliability performance, which can be visualized via metric CRPS.

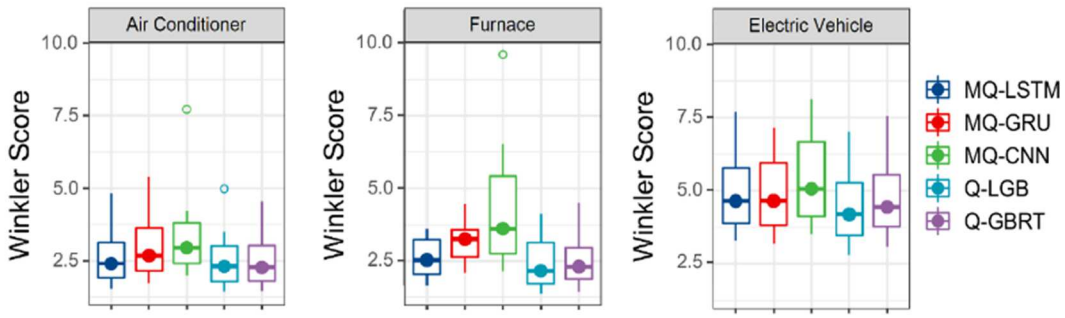


Figure 8-9. Boxplot of the Winkler Score.

From Table 8-4, by comparing the CRPS of each model for different PINC, A smaller value of CRPS indicates a better performance of the probabilistic model. It is noticed

that the CRPS value of the proposed MQ-LSTM is the smallest among most cases and PICPs. The MQ-LSTM model achieves the best performance in disaggregating AC load components. The CRPS of MQ-LSTM reduced by 37.25%, 31.91%, 28.89%, and 5.88%, compared to the values of CRPS of Q-GBRT, Q-LGB, MQ-CNN, and MQ-GRU models, respectively. The superior of MQ-LSTM is also obvious when disaggregating Furnace load components. In the furnace case, The CRPS of MQ-LSTM reduced by 25.80%, 25.00%, 11.54% and 5.48%, compared to the values of CRPS of Q-GBRT, Q-LGB, MQ-CNN, and MQ-GRU models, respectively. However, when it comes to the EV case, the performance of Q-LGB turns better than MQ-LSTM, as the CRPS value of the Q-LGB model reaches 0.068, which obtains the best performance among all probabilistic models.

Figure 8-10 presents the PIs evaluated by the proposed MQ-LSTM model with different confidence levels and the actual appliance load curve (AC, Furnace, and EV loads). As for AC load, almost all actual curves are between the upper and lower bounds. The solid yellow line, which represents the median estimation, tracks the variation of AC demand load precisely. Meanwhile, the width of PIs, especially 95% PI, are the smallest among all appliance loads. The estimation of the Furnace load is quite similar to the AC components, as the Furnace in Austin, Texas, area plays dual roles: heating and circulating air during the usage of AC. Hence, the demand load curve of the furnace is correlated to the curve of AC demand. Most of the ground truth data are within estimated PIs; the median curve of PIs overlaps with the actual furnace load curves. However, the width of PIs is wider than the PIs of estimated AC loads. This result demonstrates that the reliability of the MQ-LSTM model on Furnace load has equal performance on AC loads, but the model's sharpness on Furnace is not as good as it is on AC loads. As for EV loads, the component is harder to be separated

from the overall demand load for two reasons: Firstly, the portion of EV load is relatively small compared to the portion of AC load or Furnace load; secondly, the average operation duration of EVs is also shorter than other cases. From Figure 8-10, the PIs estimated by MQ-LSTM are compared with the actual EV load curve; the figure shows that although the model cannot estimate the exact load curve, most all operation durations are estimated precisely.

Table 8-4. Probabilistic estimation performance.

PINC (%)	Appliance	Metrics	MQ-LSTM	MQ-GRU	MQ-CNN	Q-LGB	Q-GBRT
Overall performance	AC	CRPS (MW)	0.096	0.102	0.135	0.141	0.153
	Furnace	CRPS (MW)	0.069	0.073	0.078	0.092	0.093
	EV	CRPS (MW)	0.073	0.076	0.127	0.068	0.126
95%	AC	PICP (%)	95.23	96.35	94.44	80.28	88.77
		WS	5.81	5.39	7.71	4.99	4.56
		AACE (%)	0.20	1.35	0.56	0.14	6.23
	Furnace	PICP (%)	97.80	89.93	67.60	82.00	89.20
		WS	3.61	3.51	9.61	4.12	3.57
		AACE (%)	2.80	5.07	27.40	13.00	5.73
	EV	PICP (%)	97.00	97.33	98.40	89.40	90.40
		WS	7.15	7.69	8.12	7.02	7.56
		AACE (%)	2.00	2.33	3.40	5.60	4.60
85%	AC	PICP (%)	85.46	82.47	80.71	78.66	83.33
		WS	3.57	4.05	4.22	3.48	3.54
		AACE (%)	0.46	2.53	4.29	6.34	1.67
	Furnace	PICP (%)	79.56	72.76	47.92	53.89	59.11
		WS	3.48	4.46	6.53	3.96	4.49
		AACE (%)	5.44	12.24	37.08	31.11	25.89
	EV	PICP (%)	84.29	78.24	80.94	79.44	78.21
		WS	6.29	6.55	7.41	6.32	6.91
		AACE (%)	0.71	6.76	4.06	5.56	6.79
70%	AC	PICP (%)	72.56	65.38	62.88	51.37	57.37
		WS	2.81	3.24	3.48	2.70	2.67
		AACE (%)	2.56	4.62	7.12	18.63	12.63
	Furnace	PICP (%)	62.38	52.29	32.80	36.74	46.82
		WS	3.04	3.59	4.57	2.77	2.80
		AACE (%)	7.62	17.71	37.20	33.26	23.18
	EV	PICP (%)	67.82	55.06	72.87	51.36	59.60
		WS	5.36	5.45	5.86	4.99	5.12
		AACE (%)	2.18	14.94	2.87	18.64	10.40
40%	AC	PICP (%)	42.43	36.58	36.97	25.94	35.86
		WS	2.05	2.29	2.56	1.94	1.90
		AACE (%)	2.49	3.42	3.03	14.06	4.14
	Furnace	PICP (%)	35.53	25.66	15.76	22.27	23.85
		WS	2.21	2.83	2.98	1.85	1.08
		AACE (%)	4.47	14.34	24.24	17.73	16.15
	EV	PICP (%)	29.88	20.83	26.40	34.46	24.46
		WS	4.09	4.05	4.33	3.77	4.02
		AACE (%)	10.12	19.17	13.60	5.54	15.54

The probability density curves obtained by the proposed MQ-LSTM are presented in Figure 8-11. The actual values of different load components during the time are investigated, the grey shading curves are the probability density function (PDF) with 95% confidence levels, and the red vertical line is the actual value of a specific hour, while the black dash vertical line shows the maximum probability point of the probability density curve. First, all selected actual values are in the middle of the PDFs for AC and Furnace load components, the maximum probability points almost overlap with the actual values, except for 1 pm and 2 pm, and the model has the highest accuracy when estimating the maximum and minimum values of the target loads. However, the PDF graph for EV load is not in good shape, as the maximum probability point of the PDF is not as high as expected, and the shape of the PDF is not strictly following the Gaussian distribution.

Figure 8-12 shows the estimation results of all quantile regression models for three load components from 23rd September 2018 to 28th September 2018. The figure provides a clearer view of the performance of the individual energy disaggregation model. The colour shadings are the estimated PIs between 90-quantile and 10-quantile, and the solid red curve is the ground truth curve of the load component, while other colour solid curves are the median values of estimated PIs. The figure shows that the ground truth curve is within the PIs of all models, and the PIs of MQ-LSTM and MQ-GRU can track the fluctuations of the actual values with high accuracy. Among all PIs, the PIs provided by the MQ-CNN have the most significant interval, while the widths of Q-LGB and Q-GBRT are relatively small, but the errors between the PIs of these two models and the actual values are also considerably higher. To summarise, the proposed MQ-LSTM energy disaggregation algorithm has superior performance in reliability and sharpness for all load components investigated in this chapter.

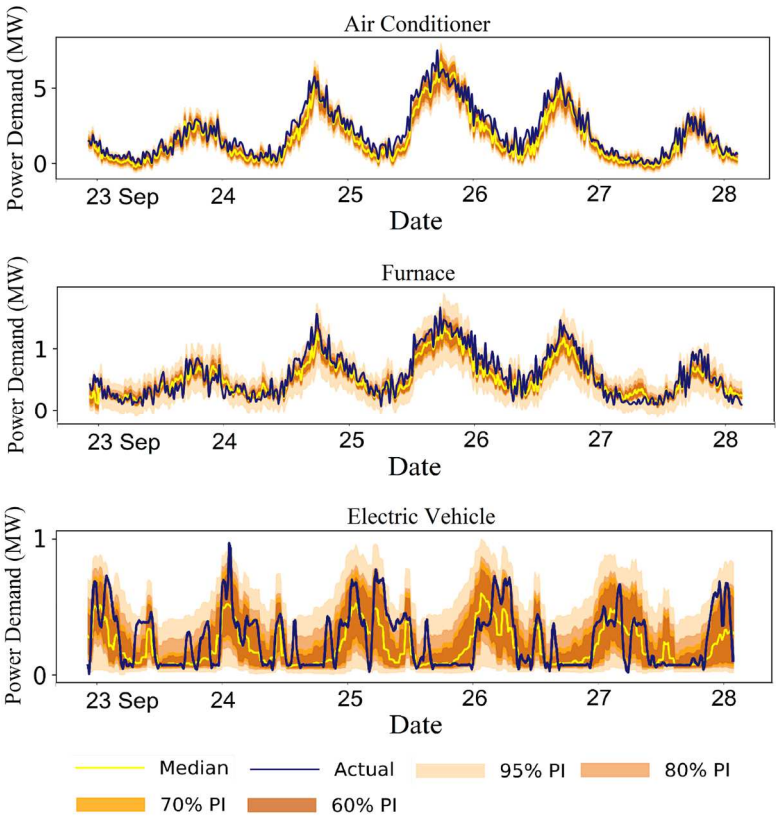


Figure 8-10. PIs of the MQ-LSTM energy disaggregation model with various confidence levels for AC, Furnace, and EV loads.

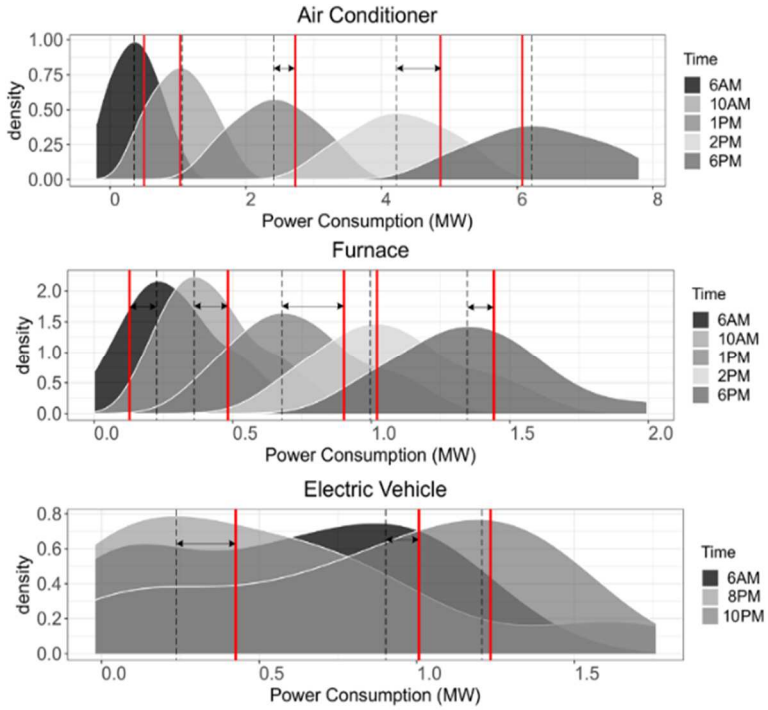


Figure 8-11. The MQ-LSTM energy disaggregation model obtained probability density curves for AC, Furnace, and EV loads.

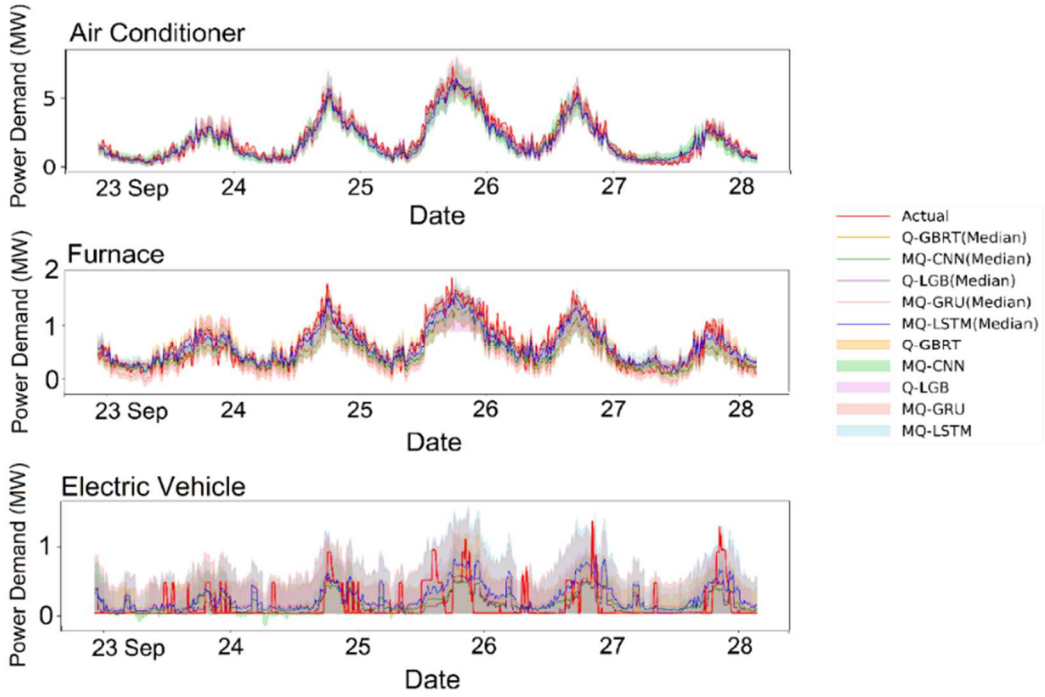


Figure 8-12. Comparison of the results of energy disintegration implemented by various algorithms. The solid curve is the median estimate, the colour shading is the range between the estimated curve of quantiles 10 and 90, while the solid red curve is the ground truth load.

8.5.3 Case study II: transferability of the proposed scheme

One major issue of the proposed energy disaggregation method is that it requires historical feeder-level demand load data to train the machine learning/ neural network models before adopting them for industrial application. However, such data is not always available in many areas, limiting a broader application. Hence, it is essential to investigate the transferability of the proposed energy disaggregation model. The transfer learning process is shown in Figure 8-13, as a transductive transfer learning problem (source data labels are available, but target data labels are unavailable [408]), the deep neural network model is pre-trained with data in the source domain, the difference between the distributions of the source domain and target domain can be minimized by modified the source domain (adjusting the portions of different load components, adjusting seasonal and trend, etc.). The source data is also normalized and resampled to fit the target data. Then the model is fine-tuned with the training set of the target domain, and the model is tested with the testing set of the target domain.

In this case study, the semi-synthetic dataset constructed from Austin, Texas data of Dataport between 2018 and 2019 (sampling frequency is 1 Min) is used as the public training dataset, and the New York data from the Dataport during 2019 (sampling frequency is 15 Min) is used as the local data to be disaggregated (testing dataset). Since the geographical location, sampling frequency, climate condition, and portion of load components are different, the two datasets can be viewed differently, and the result would be reliable. An investigation of the installation rate of AC, Furnace, and EV in the local area is implemented to roughly obtain the portion of different load components. Referring to the investigating result, the training dataset is adjusted to make the training dataset more like the testing data. Like Case Study 1, the solar energy component is separated at the first stage and followed by the demand load components to be disaggregated from the overall demand load.

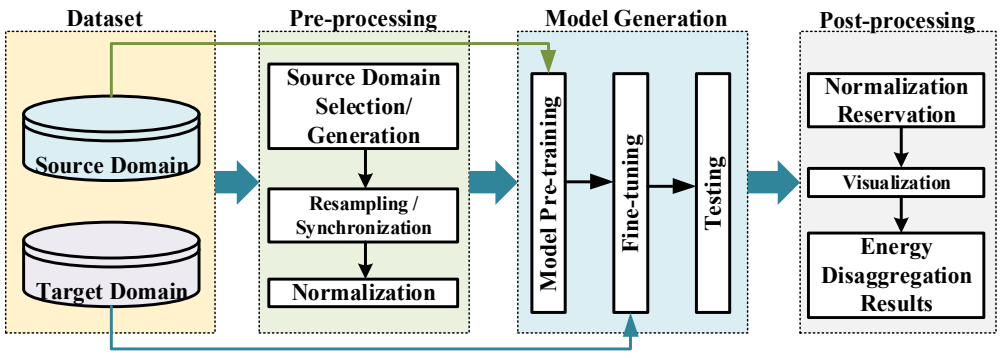


Figure 8-13. Block diagram of the transfer learning process.

Table 8-5 presents the performance metrics of the transfer learning model. Compared to the results concluded in Table 8-4, the results of the transfer learning model are not as good as the typical energy disaggregation model. The WS, AACE, and Score of the transfer learning case are more significant than the results in Case Study I. However, referring to Figure 8-15, the probabilistic models can still provide precise estimation results as most of the actual values are within the estimated PIs with narrow widths. The results show that the proposed MQ-LSTM-based energy disaggregation model has good transferability; by selecting a suitable source domain, the model can be well-trained and adopted to other feeder models.

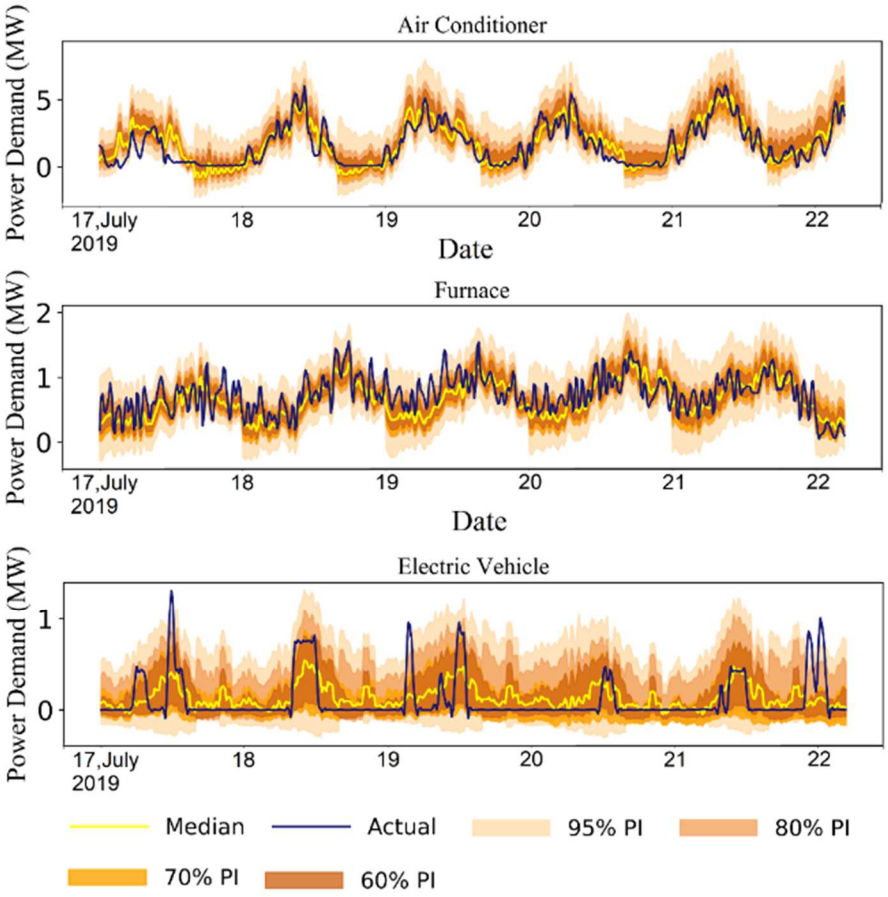


Figure 8-14. PIs of the transfer learning model with various confidence levels for AC, Furnace, and EV loads.

8.5.4 Application of Energy Disaggregation Technology in Power System

The disaggregated components obtained by the proposed probabilistic model have multiple applications in the power system industry, especially demand response and power system operation. Detailed applications are illustrated as follows:

1) Estimating the behind-the-meter renewable energy, especially residential PV energy, will increase the visibility of the power system and help the power system operator better understand the real-time condition. As a result, the operators can better deal with the stability problems raised by integrating renewable energy and better plan energy reserves [191].

Table 8-5. Performance of transfer learning.

PINC (%)	Appliance	Metrics	MQ-LSTM
Overall performance	AC	CRPS (MW)	0.154
	Furnace	CRPS (MW)	0.121
	EV	CRPS (MW)	0.178
95%	AC	PICP (%)	91.55
		WS	10.97
		AACE (%)	3.45
	Furnace	PICP (%)	89.35
		WS	16
		AACE (%)	5.65
	EV	PICP (%)	97.92
		WS	8.61
		AACE (%)	12.94
85%	AC	PICP (%)	82.18
		WS	6.97
		AACE (%)	2.82
	Furnace	PICP (%)	80.61
		WS	12.83
		AACE (%)	4.39
	EV	PICP (%)	88.54
		WS	6.85
		AACE (%)	3.54
70%	AC	PICP (%)	60.65
		WS	5.45
		AACE (%)	9.35
	Furnace	PICP (%)	62.85
		WS	8.17
		AACE (%)	7.15
	EV	PICP (%)	59.99
		WS	5.65
		AACE (%)	10.01
40%	AC	PICP (%)	43.17
		WS	3.63
		AACE (%)	3.17
	Furnace	PICP (%)	26.39
		WS	5.82
		AACE (%)	13.61
	EV	PICP (%)	32.7
		WS	3.88
		AACE (%)	7.3

2) Demand response programs that involve TCLs can reduce the 10-30% peak loads in most cases [409]. TCLs, especially AC and furnace loads, can be cut down or turned off for a short period without influencing the customer's normal life. Hence, the demand response operator can better plan demand response actions by knowing TCL portions.
3) The demand response operator can better optimize capacity bids in electricity markets by real-time estimating demand-responsive loads.
4) The estimation of the demand-responsive loads can be used as a feedback signal in the load coordination algorithms [395].

8.5.5 Limitation of the method

Although the proposed feeder-level energy disaggregation method achieves good accuracy in separating load compositions, the limitation of this work is summarized as follows: The proposed method has a national bias and highly depends on the nation/area of the dataset used for training. The appliance categories change a lot in different countries and areas; for instance, the AC, widely installed in the U.S. and China, is not common in the U.K. The proposed transfer learning approach can only be transferred to an area with similar load components.

To better interpret the proposed model in other countries, such as the U.K., several approaches can be employed:

1) Build a high-resolution household-level electricity dataset with detailed appliance usage readings for the U.K., then use the data to build distribution network models. Existing U.K. datasets such as Low Carbon London [410] can only provide household-level electricity consumption, while appliance-level data is unavailable.
2) Develop a more flexible unsupervised/semi-supervised learning model which can adapt to the distribution networks in different countries and areas.

8.6 Privacy Risk Analysis

In this section, the privacy risk of the proposed feeder-level energy disaggregation method is analysed by comparing it with the existing method. As shown in Figure 8-15. The existing approach to obtaining the feeder-level load components requires first obtaining load components at each house respectively. The DNO need to implement the NILM data mining algorithm on the individual smart meter readings to infer detailed appliance consumption (Appliance *1*-*N* in Figure 8-15), and then the DNO will sum up the consumption of the same appliance for all houses to determine the portion of this load category under the feeder. Such an approach is complex and

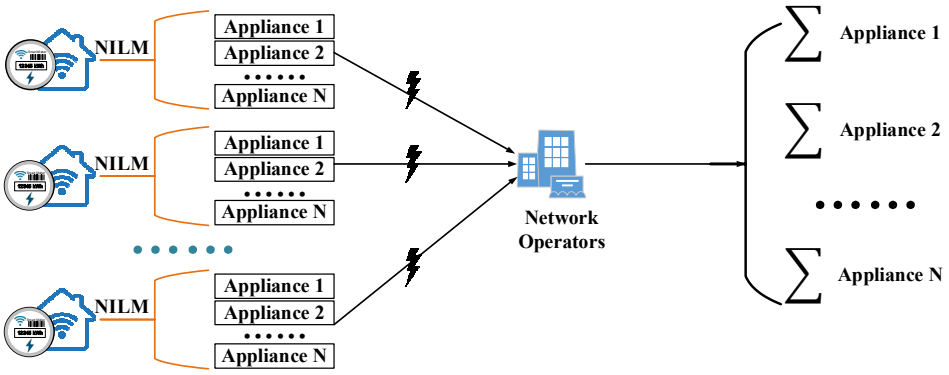
introduces various privacy risks. Detailed risks and disadvantages can be summarized as follows:

1) As demonstrated in Chapter 4, high-frequency appliance-level data links a large amount of personal information, which could reveal detailed behaviour patterns of the energy consumers.
2) The existing approach requires DNO access to an individual's smart meter data, which contradicts the data access policy of OFGEM [41] and BEIS [12].
3) By sharing the appliance-level data with the DNO, external adversaries can obtain the information that may eavesdrop on the communication between the smart meter and the DNO.
4) This method requires all smart meters in the distribution network to participate; the absence of a part of meters would result in a mismatch between the estimation and the actual value.

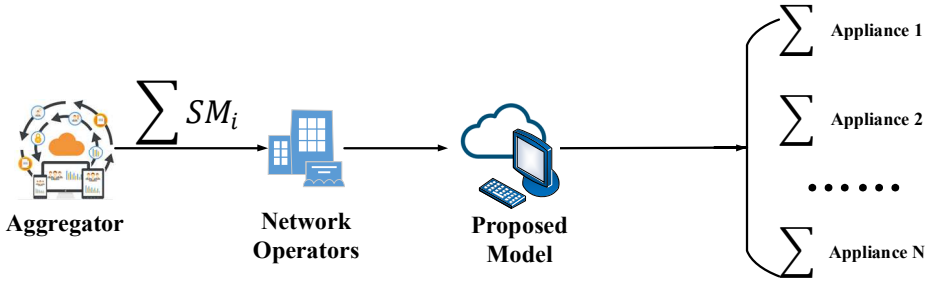
As for the proposed feeder-level energy disaggregation method, the NILM data mining algorithm on individual electricity consumption is not required. Only the aggregated readings provided by the aggregator are shared with the DNO, and the DNO has no authority to access the individual meter readings. In such case:

1) Load components algorithm moves from existing household-level to feeder/distribution level, which decreases the possibility that the adversary invades personal information.
2) Given the aggregated data and the external information, the proposed model determines the portion of each load component from the aggregated demand. The proposed scheme strictly follows the data access regulation made by OFGEM [41] and BEIS [12]; both the input and output of the proposed model are aggregated data without accessing any individual information.
3) The external adversary has no/less opportunity to obtain the individual consumption/appliance information as such information has never been shared.
4) Furthermore, the proposed model is more reliable than the existing method as only the aggregators, and all smart meters are required to participate.

Based on the analysis above, the privacy risk in the current feeder-level energy disaggregation model is reduced.



(a) The existing approach to obtain load components at the feeder level.



(a) The proposed approach is to obtain load components at the feeder level.

Figure 8-15. Comparison of the existing/proposed feeder level energy disaggregation methods.

8.7 Chapter Summary

Understanding the load components under the grid supply point will contribute to the utility's ability to deliver improved demand-side management and peak shaving services. This chapter proposes a probabilistic feeder-level energy disaggregation model based on MQ-LSTM deep neural network. The purpose of the proposed model is to disaggregate the net load into four components: renewable energy generation (rooftop PVs in this case), TCLs (AC and Furnace loads), and Non-TCLs (EV load chosen as a case study), and other loads.

Then other load components are disaggregated via the proposed MQ-LSTM algorithm; the proposed model considers various relevant variables as input features, including current and historical demand load and temperature, meteorological measurement, and calendar information. Four state-of-the-art probabilistic machine learning/ deep learning models: MQ-GRU, MQ-CNN, Q-LGB, and Q-GBRT algorithms, are adopted as the benchmarks. Two case studies thoroughly investigate the performance of the proposed model and benchmark models in four aspects: reliability (PICP, AACE), sharpness (WS), training time and overall performance (CRPS). The case studies confirm that the proposed model performs better in disaggregating different load components. As for AC and Furnace loads, the proposed model can provide reliable and sharp PIs and estimate the precise load curves. Although the detailed load curve is not available when it turns to EV load, the probabilistic model can estimate the intervals with high accuracy.

Moreover, the transferability of the proposed model is studied as well since obtaining a large amount of labelled data for training is unrealistic. The model is pre-trained with a public dataset (source domain) and then tested with local data (target domain). The results show that the proposed model is transferable to a different area with different interval resolutions and different portions of load components.

Part IV Conclusion

Chapter 9 Conclusion and Future Work

9.1 Conclusion

This thesis proposes a multi-channel smart metering system that fulfils reasonable and ethical user and system functionality whilst the energy consumer's privacy is guaranteed. The thesis starts with investigating the configurations of the current smart metering systems and the limitations of the current solutions to the privacy leakages. Then, based on the system's vulnerability, a threat/adversary model is developed to utilize efficient data mining techniques to infer personal information. To defend the smart meter data against the threat/adversary model and better align to the GDPR, a multi-channel smart metering system is developed to transmit different granularity data between the consumers and other stakeholders (energy supplier, distribution operator, and third parties). By comparing the privacy risks between the proposed and the existing systems, the conclusion is made that the privacy risks raised by the threat/adversary are reduced after the mitigations proposed.

Then based on the proposed smart metering system, functionalities required by the third-party service provider and the distribution network operator are validated via case studies. The federated learning-based value-added service platform provides an edge-cloud computing infrastructure to enable the smart meter to be analysed locally, and the TP, the honest-but-curious adversary, is prevented from accessing the smart meter data. Whilst the DNO benefits from the physical/informatic aggregator to improve the predictability (short-term load forecasting), the visibility (feeder-level energy disaggregation) and reduce the uncertainty (solar energy decoupling) of the distribution network. The results validate that the proposed system can better benefit distribution network operation without scarifying privacy.

Seven original contributions and finds of the thesis are summarized as follows:

9.1.1 A comprehensive attacker/threat model

The threat/adversary model with the motivation to obtain personal data from the smart metering system is defined at the beginning of the thesis. The purposes and the routes for the adversary to obtain personal information are studied. This thesis includes the inner adversary (third-party service provider) and the external adversary (adversary who can eavesdrop on the communication channels). The proposed smart metering is better assessed by developing the threat/adversary model.

9.1.2 A multi-channel smart metering system

A multi-channel smart metering system is developed based on the threat/adversary model, the requirements from the GDPR, and the compulsory functions required by the stakeholders. The core strategy of the proposed system is only to transmit the minimum granularity data to the stakeholders to reduce the possibility that the personal data to be disclosed. In the proposed system, three channels are developed; the first is the high-frequency data aggregation channel, which utilizes physical/informatic aggregators, collects the readings from the neighbouring smart meters, and only shares the aggregated data to the energy utility. The second channel is the TOU billing channel, and the bills are generated by the smart meter and only the cumulative energy consumption and bills during a reporting duration are sent to the ES.

9.1.3 The privacy boundary of the smart meter data

The boundary between sensitive and insensitive smart meter data is not well studied in the literature. A NILM-based data mining algorithm used by the adversary is employed to detect the privacy boundary of two vital parameters: the aggregation size and the interval resolution. By evaluating the detectability of the NILM algorithm on different granularity datasets, a three-level privacy boundary benchmark is concluded. The results show that the data with an aggregation size over 40 or an interval

resolution over eight hours can provide complete protection to consumers' private information.

9.1.4 A federated learning platform to enable third-party value-added services

As the potential internal adversary in the system, the third-party service providers should be restricted from accessing the personal data while the accuracy of the services should not be influenced. To better solve the contradiction, a differential private federated learning platform is developed. The FL provides a strong privacy guarantee to the clients, and the client's data will not leave their homes during the process. This way, the probability of sensitive information being inferred or used by potential adversaries is reduced. Consumers could be more willing to share their data with third parties if this is the case. Moreover, the modular design allows the platform to access external databases and provide multiple TPSs.

9.1.5 A distribution level load forecasting method with aggregated smart meter data

The distribution network was not well monitored in the past due to the lack of proper metering infrastructure and communication channels. The proposed smart metering system provides high-quality aggregated smart meter data that can help the DNO better forecast load without accessing the individual consumptions. This thesis introduced a hybrid STLF method to predict day-ahead load at the distribution level. Unlike the traditional machine learning/deep learning method, the proposed method extracts both frequency and time domain features from the netload, and the results show that the proposed hybrid STLF achieves more precise prediction results.

9.1.6 A solar energy decoupling method at the grid supply point

Behind-the-meter PV generation brings uncertainty for the DNO to monitor the distribution network; accessing each PV meter installed at the consumer's house

increases the privacy risk and is time-consuming. The proposed smart metering system provided real-time netload data of the distribution feeder or the distribution network from the physical/informatic aggregator. Three solar energy decoupling models (unsupervised upscaling model, supervised GBRT model, and 1D CNN-LSTM model) are built with the netload data. The proposed model can separate the PV generation from the netload at the grid supply point on a real-time base, given external information such as irradiance data and weather conditions.

9.1.7 A Probabilistic energy disaggregation method at the feeder's head

The load components under the feeder/LV distribution network are important information for the DNO to design a demand response plan. In the current smart metering system, such information can only be obtained by accessing an individual's smart meter readings and implementing the NILM algorithm to each data. Such access is in contradicts the BEIS specification and increases the privacy risks. A probabilistic feeder-level energy disaggregation model is developed based on the proposed smart metring system to disaggregate the measured netload into controllable and uncontrollable loads with the aggregated data provided by the physical/informatic aggregator only. Moreover, a transfer learning approach is proposed to increase the flexibility and robustness of the energy disaggregation model.

9.2 Plan to Influence Existing Industry Specification

The findings and contributions of this thesis will also benefit the policy marker and can potentially influence the existing industry specifications.

1) Although the existing Review of the Data Access and Privacy Framework published by BEIS highlights that the data should be aggregated before being shared with the DNO, the aggregation size is not quantified in the current

document. With the privacy boundary quantified by the thesis, BEIS can better specify the granularity of the data that can be shared with different stakeholders.

2) As discussed in the thesis, the smart meter data contains personal information and should be regulated by the GDPR. However, the existing smart metering system does not follow the GDPR strictly, as no option is given to consumers to select the granularity of data to be transmitted. The further generation of SMETS is expected to send multiple resolution data to different entities, which follows the data minimization principle.
3) Referring to the results summarised in this thesis, policymakers, especially BEIS and OFGEM, should carefully implement methods such as the noise-adding method. Although these methods would reduce the sensitivity of personal information and thus risks of privacy intrusion, the data's usability and value would decrease, potentially undermining the achievement of benefits for stakeholders.
4) Existing SMETS 2 focuses on billing and recording, while the next generation smart meter requires higher computation ability and storage space to enable smart home management and interaction with the smart appliances.

9.3 Future work

Although a hierarchical smart metering system is designed to illustrate the privacy and functionality configurations, there are still some limitations of the existing works and several research problems to be settled in future works.

9.3.1 Tamper-resistance of smart meters

The smart metering system proposed in this research assumes that the smart meter is tamper-resistance to guarantee the correctness of the reading. The real-world smart meter could be tempered by the attacker and would bypass any data protection technologies. Future work should develop an economical solution to guarantee the integrity of the smart meters.

9.3.2 The influence of the smart appliances on the privacy

IoT-based smart appliances are becoming more intelligent with the help of AI, and their domestic adoption continues to soar. However, before the widespread adoption of beneficial technologies like Energy Demand Side Management (DSM), future work is required to understand how the interactions between multiple smart devices in the home can affect home users' privacy, security, and safety.

9.3.3 Security of the value-added service platform

Although the federated learning model is efficient in training cloud model via the edge devices, it is vulnerable to suffering from the cyberattack, as a significant abnormal data from a client while cause to fail the whole cloud server. The malicious clients could be a fake smart meter that hackers clone or be controlled by an attacker who wants to destroy the third-party platform. Hence, future work is required to provide better security to the services.

9.3.4 The influence of the smart meter and rooftop PV on the human behaviours

The installation of the smart meter and the rooftop PV improve consumers' energy awareness and reshape their lifecycle. Peoples are more willing to use their appliances on a sunny day when more solar energy is generated or during the off-peak period with low tariffs. Future work should investigate how these devices benefit the DSM programmes.

9.3.5 Reduce the national bias of the methodology

The existing methodology and the datasets proposed in this thesis have a national bias, and the results of the experiments highly depend on specific countries/ areas. Due to the limitation of public datasets and laboratory/facilities, existing work only focuses on the appliances in the US. Hence, future work can be conducted in the following

aspects to eliminate/ reduce the national bias: (1) extend the existing database to include more appliances and smart meter data from different counties; (2) evaluate the proposed method with the energy data from different countries to obtain the results which fit specific countries.

References

[1] K. L. Lueth. "Smart Meter Market 2019: Global penetration reached 14% – North America, Europe ahead," 16th, Jun, 2021; https://iot-analytics.com/smart-meter-market-2019-global-penetration-reached-14-percent/.

[2] N. J. King, and P. W. Jessen, "Smart metering systems and data sharing: why getting a smart meter should also mean getting strong information privacy controls to manage data sharing," *International Journal of Law and Information Technology,* vol. 22, no. 3, pp. 215-253, 2014.

[3] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building, 2010, pp. 61-66.

[4] E. Quinn, "Smart Metering and Privacy: Existing Laws and Competing Policies," *SSRN Electronic Journal*, 05/09, 2009.

[5] M. Autili, D. Di Ruscio, P. Inverardi, P. Pelliccione, and M. Tivoli, "A software exoskeleton to protect and support citizen's ethics and privacy in the digital world," *IEEE Access,* vol. 7, pp. 62011-62021, 2019.

[6] C. Taylor, *The ethics of authenticity*: Harvard University Press, 1992.

[7] P. Voigt, and A. Von dem Bussche, *The eu general data protection regulation (gdpr): A Practical Guide*, 1st ed.: Springer International Publishing, 2017.

[8] L. Floridi, "Soft ethics and the governance of the digital," *Philosophy Technology,* vol. 31, no. 1, pp. 1-8, 2018.

[9] L. Introna, and A. Pouloudi, "Privacy in the information age: Stakeholders, interests and values," *Journal of Business Ethics,* vol. 22, no. 1, pp. 27-38, 1999.

[10] M. Foucault, *The Foucault effect: Studies in governmentality*: University of Chicago Press, 1991.

[11] J. Rachels, "Why privacy is important," *Privacy*, pp. 11-21: Routledge, 2017.

[12] "SMART METERING IMPLEMENTATION PROGRAMME: Review of the Data Access and Privacy Framework," E. I. S. Department for Business, UK, ed., 2018.

[13] L. Hernández-Callejo, "A Comprehensive Review of Operation and Control, Maintenance and Lifespan Management, Grid Planning and Design, and Metering in Smart Grids," *Energies,* vol. 12, no. 9, pp. 1630, 2019.

[14] M. Kerai, *Smart Meter Statistics in Great Britain: Quarterly Report to end December 2020*, Department for Business, Energy & Industrial Strategy (BEIS), 2021.

[15] DECC, *Smart Metering Implementation Programme: Smart Metering Equipment Technical Specifications Version 1.58*, Department of Energy and Climate Change, 2014.

[16] N. Wan, and K. Manning, "Exceeding 60-year life expectancy from an electronic energy meter," in Proceedings Metering Asia Pacific Conference, 2001, pp. 20-22.

[17] K. Sharma, and L. M. Saini, "Performance analysis of smart metering for smart grid: An overview," *Renewable and Sustainable Energy Reviews,* vol. 49, pp. 720-735, 2015.

[18] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen, and X. Shen, "Energy-theft detection issues for advanced metering infrastructure in smart grid," *Tsinghua Science and Technology,* vol. 19, no. 2, pp. 105-120, 2014.

[19] T. D. Tamarkin, "Automatic meter reading," *Public Power,* vol. 50, no. 5, pp. 934-937, 1992.

[20] K. Weranga, S. Kumarawadu, and D. Chandima, "Evolution of electricity meters," *Smart Metering Design and Applications*, pp. 17-38: Springer, 2014.

[21] "Smart Metering Implementation Programme: Communications Hub Technical Specifications Version 1.46," D. o. E. a. C. Change, ed., Department of Energy and Climate Change, 2014.

[22] L.-H. Yen, and W.-T. Tsai, "The room shortage problem of tree-based ZigBee/IEEE 802.15. 4 wireless networks," *Computer Communications,* vol. 33, no. 4, pp. 454-462, 2010.

[23] C. Stanton, *The guide to Load Switching for Smart Electricity Meter Manufacturers*, REL Developments, 2015.

[24] S. Darby, C. Liddell, D. Hills, and D. Drabble, "Smart metering early learning project: synthesis report," 2015.

[25] M. Pullinger, H. Lovell, and J. Webb, "Influencing household energy practices: a critical review of UK smart metering standards and commercial feedback devices," *Technology Analysis & Strategic Management,* vol. 26, no. 10, pp. 1144-1162, 2014.

[26] M. Kochański, K. Korczak, and T. Skoczkowski, "Technology Innovation System Analysis of Electricity Smart Metering in the European Union," *Energies,* vol. 13, no. 4, pp. 916, 2020.

[27] M. Fahim, and A. Sillitti, "Analyzing load profiles of energy consumption to infer household characteristics using smart meters," *Energies,* vol. 12, no. 5, pp. 773, 2019.

[28] A. Llano, I. Angulo, D. de la Vega, and L. Marron, "Virtual PLC Lab Enabled Physical Layer Improvement Proposals for PRIME and G3-PLC Standards," *Applied Sciences,* vol. 10, no. 5, pp. 1777, 2020.

[29] A. Braeken, P. Kumar, and A. Martin, “Efficient and Provably Secure Key Agreement for Modern Smart Metering Communications,” *Energies,* vol. 11, no. 10, pp. 2662, 2018.

[30] A. Karale, “The Challenges of IoT addressing Security, Ethics, Privacy and Laws,” *Internet of Things*, pp. 100420, 2021.

[31] E. McKenna, I. Richardson, and M. J. E. P. Thomson, “Smart meter data: Balancing consumer privacy concerns with legitimate applications,” vol. 41, pp. 807-814, 2012.

[32] K. C. Budka, J. G. Deshpande, T. L. Doumi, M. Madden, and T. Mew, “Communication network architecture and design principles for smart grids,” *Bell Labs Technical Journal,* vol. 15, no. 2, pp. 205-227, 2010.

[33] M. Kuzlu, M. Pipattanasomporn, and S. Rahman, “Communication network requirements for major smart grid applications in HAN, NAN and WAN,” *Computer Networks,* vol. 67, pp. 74-88, 2014.

[34] E. Webborn, S. Elam, E. McKenna, and T. Oreszczyn, “Utilising smart meter data for research and innovation in the UK,” in ECEEE summer study, 2019, pp. 1387-1396.

[35] *DCC Business & Development Plan (2021/22 - 2025/26)*, The Data Communications Company (DCC) 2021.

[36] L. Thomas, and N. Jenkins, “Smart metering for the UK,” 2012.

[37] U. Doe, “An assessment of energy technologies and research opportunities,” *Quadrennial Technology Review. United States Department of Energy*, pp. 12-19, 2015.

[38] C. Payne, “The commercial energy consumer: About whom are we speaking?,” 2006.

[39] J. Zarnikau, and I. Hallett, “Aggregate industrial energy consumer response to wholesale prices in the restructured Texas electricity market,” *Energy Economics,* vol. 30, no. 4, pp. 1798-1808, 2008.

[40] S. C. Vegunta, C. F. A. Watts, S. Z. Djokic, J. V. Milanović, and M. J. Higginson, “Review of GB electricity distribution system's electricity security of supply, reliability and power quality in meeting UK industrial strategy requirements,” *IET Generation, Transmission & Distribution,* vol. 13, no. 16, pp. 3513-3523, 2019.

[41] "Open letter on DNOs’ privacy plans for access to smart meter data," U. The Office of Gas and Electricity Markets (Ofgem), ed., 2016.

[42] Ofgem, "Energy Supply Probe–Initial Findings Report," Ofgem London, 2008.

[43] R. C. Parks, “Advanced metering infrastructure security considerations,” *SANDIA REPORT, Sandia National Laboratories*, 2007.

[44] J. McCullough, “AMI security considerations,” *Elster*, 2010.

[45] M. R. Asghar, G. Dán, D. Miorandi, and I. Chlamtac, "Smart meter data privacy: A survey," *IEEE Communications Surveys & Tutorials,* vol. 19, no. 4, pp. 2820-2835, 2017.

[46] C. A. C. Montañez, and W. Hurst, "A Machine Learning Approach for Detecting Unemployment Using the Smart Metering Infrastructure," *IEEE Access,* vol. 8, pp. 22525-22536, 2020.

[47] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. García, "NILM Techniques for Intelligent Home Energy Management and Ambient Assisted Living: A Review," *Energies,* vol. 12, pp. 2203, 06/10, 2019.

[48] A. C.-F. Chan, and J. Zhou, "On smart grid cybersecurity standardization: Issues of designing with NISTIR 7628," *IEEE Communications Magazine,* vol. 51, no. 1, pp. 58-65, 2013.

[49] G. Eibl, and D. Engel, "Influence of Data Granularity on Smart Meter Privacy," *IEEE Transactions on Smart Grid,* vol. 6, pp. 1-1, 12/18, 2014.

[50] C. Beckel, L. Sadamori, T. Staake, and S. Santini, "Revealing household characteristics from smart meter data," *Energy,* vol. 78, pp. 397-410, 2014.

[51] J. L. Viegas, S. M. Vieira, and J. Sousa, "Mining consumer characteristics from smart metering data through fuzzy modelling," in International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, 2016, pp. 562-573.

[52] G. Sun, Y. Cong, D. Hou, H. Fan, X. Xu, and H. Yu, "Joint household characteristic prediction via smart meter data," *IEEE Transactions on Smart Grid,* vol. 10, no. 2, pp. 1834-1844, 2017.

[53] S. Sultan, "Privacy-preserving metering in smart grid for billing, operational metering, and incentive-based schemes: A survey," *Computers & Security,* vol. 84, pp. 148-165, 2019/07/01/, 2019.

[54] G. Giaconi, D. Gunduz, and H. V. Poor, "Privacy-Aware Smart Metering: Progress and Challenges," *IEEE Signal Processing Magazine,* vol. 35, no. 6, pp. 59-78, 2018.

[55] F. Farokhi, and H. Sandberg, "Fisher information as a measure of privacy: Preserving privacy of households with smart meters using batteries," *IEEE Transactions on Smart Grid,* vol. 9, no. 5, pp. 4726-4734, 2017.

[56] O. Tan, J. Gómez-Vilardebó, and D. Gündüz, "Privacy-cost trade-offs in demand-side management with storage," *IEEE Transactions on Information Forensics Security,* vol. 12, no. 6, pp. 1458-1469, 2017.

[57] Y. Sun, L. Lampe, and V. Wong, "Smart meter privacy: Exploiting the potential of household energy storage units," *IEEE Internet of Things Journal,* vol. 5, no. 1, pp. 69-78, 2017.

[58] S. McLaughlin, P. McDaniel, and W. Aiello, "Protecting consumer privacy from electric load monitoring," in Proceedings of the 18th ACM conference

on Computer and communications security, Chicago, Illinois, USA, 2011, pp. 87–98.

[59] G. Kalogridis, C. Efthymiou, S. Z. Denic, T. A. Lewis, and R. Cepeda, "Privacy for Smart Meters: Towards Undetectable Appliance Load Signatures," in 2010 First IEEE International Conference on Smart Grid Communications, 2010, pp. 232-237.

[60] G. Giaconi, D. Gündüz, and H. V. Poor, "Optimal demand-side management for joint privacy-cost optimization with energy storage," in 2017 IEEE International Conference on Smart Grid Communications (SmartGridComm), 2017, pp. 265-270.

[61] Y. Sun, L. Lampe, and V. W. Wong, "Combining electric vehicle and rechargeable battery for household load hiding," in 2015 IEEE International Conference on Smart Grid Communications (SmartGridComm), 2015, pp. 611-616.

[62] J. Gomez-Vilardebo, and D. Gündüz, "Smart meter privacy for multiple users in the presence of an alternative energy source," *IEEE Transactions on Information Forensics Security,* vol. 10, no. 1, pp. 132-141, 2014.

[63] D. Varodayan, and A. Khisti, *Smart meter privacy using a rechargeable battery: Minimizing the rate of information leakage*, 2011.

[64] "bmz gmbh li-io ess 3.0 lithium-ionen-energy storage system 3.0 for sma," 30th, April, 2020; https://www.off-grid-europe.com/bmz-gmbh-li-io-ess-3-0-lithium-ionen-energy-storage-system-3-0-for-sma?gclid=CjwKCAjwv41BRAhEiwAtMDLsuqBrUnfvzcJRUb7IOyCkaH1XWJZAQY7XuNHR5qNVUYk5S9grA7aHxoC1qYQAvD_BwE.

[65] X. Zhao, X. Ma, B. Chen, Y. Shang, and M. Song, "Challenges toward carbon neutrality in China: Strategies and countermeasures," *Resources, Conservation and Recycling,* vol. 176, pp. 105959, 2022.

[66] D. Egarter, C. Prokop, and W. Elmenreich, "Load hiding of household's power demand," in 2014 IEEE International Conference on Smart Grid Communications (SmartGridComm), 2014, pp. 854-859.

[67] D. Chen, S. Kalra, D. Irwin, P. Shenoy, and J. Albrecht, "Preventing occupancy detection from smart meters," *IEEE Transactions on Smart Grid,* vol. 6, no. 5, pp. 2426-2434, 2015.

[68] X. He, X. Zhang, and C.-C. J. Kuo, "A distortion-based approach to privacy-preserving metering in smart grids," *IEEE Access,* vol. 1, pp. 67-78, 2013.

[69] G. Eibl, and D. Engel, "Differential privacy for real smart metering data," *Computer Science - Research and Development,* vol. 32, no. 1, pp. 173-182, 2017/03/01, 2017.

[70] G. Ács, and C. Castelluccia, "I Have a DREAM! (DiffeRentially privatE smArt Metering)," in Information Hiding, Berlin, Heidelberg, 2011, pp. 118-132.

[71] J. Bohli, C. Sorge, and O. Ugus, "A Privacy Model for Smart Metering," in 2010 IEEE International Conference on Communications Workshops, 2010, pp. 1-5.

[72] P. Barbosa, A. Brito, H. Almeida, and S. Clauß, "Lightweight privacy for smart metering data by adding noise," in Proceedings of the 29th Annual ACM Symposium on Applied Computing, 2014, pp. 531-538.

[73] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, "EPPA: An Efficient and Privacy-Preserving Aggregation Scheme for Secure Smart Grid Communications," *IEEE Transactions on Parallel and Distributed Systems,* vol. 23, no. 9, pp. 1621-1631, 2012.

[74] N. Buescher, S. Boukoros, S. Bauregger, and S. Katzenbeisser, "Two Is Not Enough: Privacy Assessment of Aggregation Schemes in Smart Metering," *Proceedings on Privacy Enhancing Technologies,* vol. 2017, 10/01, 2017.

[75] K. Kursawe, G. Danezis, and M. Kohlweiss, "Privacy-friendly aggregation for the smart-grid," in International Symposium on Privacy Enhancing Technologies Symposium, 2011, pp. 175-191.

[76] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in 2010 first IEEE international conference on smart grid communications, 2010, pp. 327-332.

[77] Y. Chen, J.-F. Martínez-Ortega, P. Castillejo, and L. López, "A homomorphic-based multiple data aggregation scheme for smart grid," *IEEE Sensors Journal,* vol. 19, no. 10, pp. 3921-3929, 2019.

[78] M. Fan, and X. Zhang, "Consortium blockchain based data aggregation and regulation mechanism for smart grid," *IEEE Access,* vol. 7, pp. 35929-35940, 2019.

[79] P. Singh, M. Masud, M. S. Hossain, and A. Kaur, "Blockchain and homomorphic encryption-based privacy-preserving data aggregation model in smart grid," *Computers & Electrical Engineering,* vol. 93, pp. 107209, 2021.

[80] S. Tonyali, K. Akkaya, N. Saputro, A. S. Uluagac, and M. Nojoumian, "Privacy-preserving protocols for secure and reliable data aggregation in IoT-enabled smart metering systems," *Future Generation Computer Systems,* vol. 78, pp. 547-557, 2018.

[81] R. Yackel, *What is homomorphic encryption, and why isn't it mainstream?*, 2021.

[82] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "TFHE: fast fully homomorphic encryption over the torus," *Journal of Cryptology,* vol. 33, no. 1, pp. 34-91, 2020.

[83] C. Thoma, T. Cui, and F. Franchetti, "Secure multiparty computation based privacy preserving smart metering system," in 2012 North American power symposium (NAPS), 2012, pp. 1-6.

[84] C. Dwork, “Differential privacy: A survey of results,” in International conference on theory and applications of models of computation, 2008, pp. 1-19.

[85] C. Efthymiou, and G. Kalogridis, “Smart grid privacy via anonymization of smart metering data,” in 2010 first IEEE international conference on smart grid communications, 2010, pp. 238-243.

[86] S. Martínez, F. Sebé, and C. Sorge, “Measuring privacy in smart metering anonymized data,” *arXiv preprint arXiv:.04863*, 2020.

[87] A. Cárdenas, S. Amin, and G. Schwartz, “Privacy-aware sampling for residential demand response programs,” *Proceedings of 1st international ACM*, 2012.

[88] F. Knirsch, G. Eibl, and D. Engel, “Multi-resolution privacy-enhancing technologies for smart metering,” *EURASIP Journal on Information Security,* vol. 2017, no. 1, pp. 6, 2017.

[89] D. L. Kehl, and S. A. Kessler, “Algorithms in the criminal justice system: Assessing the use of risk assessments in sentencing,” 2017.

[90] Y. Benkler, “Don't let industry write the rules for AI,” *Nature,* vol. 569, no. 7754, pp. 161-162, 2019.

[91] C. Cath, S. Wachter, B. Mittelstadt, M. Taddeo, and L. Floridi, “Artificial intelligence and the ‘good society’: the US, EU, and UK approach,” *Science and engineering ethics,* vol. 24, no. 2, pp. 505-528, 2018.

[92] P. Hustinx, “EU data protection law: The review of directive 95/46/EC and the proposed general data protection regulation,” *Collected courses of the European University Institute’s Academy of European Law, 24th Session on European Union Law*, pp. 1-12, 2013.

[93] H. C. Assistance, “Summary of the hipaa privacy rule,” *Office for Civil Rights*, 2003.

[94] D. Ritvo, C. Bavitz, R. Gupta, and I. Oberman, “Privacy and Children's Data-An Overview of the Children's Online Privacy Protection Act and the Family Educational Rights and Privacy Act,” *Berkman Center Research Publication*, no. 23, 2013.

[95] F. H. Cate, “The EU data protection directive, information privacy, and the public interest,” *Iowa L. Rev.,* vol. 80, pp. 431, 1994.

[96] L. de la Torre, “A guide to the california consumer privacy act of 2018,” *Available at SSRN 3275571*, 2018.

[97] J. Petters, “Data Privacy Guide: Definitions Explanations and Legislation: Varonis,” *Retrieved November,* vol. 9, pp. 2020, 2020.

[98] C. Alaton, and F. Tounquet, “Benchmarking Smart Metering Deployment in the EU-28,” *European Commission*, 2018.

[99] S. Wachter, B. Mittelstadt, and C. Russell, “Counterfactual explanations without opening the black box: Automated decisions and the GDPR,” *Harv. JL & Tech.,* vol. 31, pp. 841, 2017.

[100] R. t. t. E. Commission, *Essential regulatory requirements and recommendations for data handling, data safety, and consumer protection*, Technical Report version 1.0, 2011. https://ec. europa. eu/energy/sites/ener ….

[101] R. Camhi, and S. Lyon, “What Is the California Consumer Privacy Act?,” *Risk Management,* vol. 65, no. 9, pp. 12-14, 2018.

[102] H. F. Atlam, and G. B. Wills, "IoT security, privacy, safety and ethics," *Digital twin technologies and smart cities*, pp. 123-149: Springer, 2020.

[103] T. Hatzakis, R. Rodrigues, and D. Wright, “Smart Grids and Ethics,” *Orbit J,* vol. 2, 2019.

[104] E. McKenna, I. Richardson, and M. Thomson, “Smart meter data: Balancing consumer privacy concerns with legitimate applications,” *Energy Policy,* vol. 41, pp. 807-814, 2012/02/01/, 2012.

[105] T. Morey, T. Forbath, and A. Schoop, “Customer data: Designing for transparency and trust,” *Harvard Business Review,* vol. 93, no. 5, pp. 96-105, 2015.

[106] A. CAVOUKIAN, and S. KINGSMILL, "Privacy by Design Setting a new standard for privacy certification," Deloitte and Ryerson University, 2016.

[107] A. Senarath, and N. A. G. Arachchilage, “A data minimization model for embedding privacy into software systems,” *Computers & Security,* vol. 87, pp. 101605, 2019.

[108] M. Saltarella, G. Desolda, and R. Lanzilotti, “Privacy Design Strategies and the GDPR: A Systematic Literature Review,” in International Conference on Human-Computer Interaction, 2021, pp. 241-257.

[109] J.-H. Hoepman, “Privacy design strategies,” in IFIP International Information Security Conference, 2014, pp. 446-459.

[110] A. Cavoukian, “Privacy by design: The 7 foundational principles,” *Information and privacy commissioner of Ontario, Canada,* vol. 5, pp. 12, 2009.

[111] J.-H. Hoepman, “Privacy design strategies (the little blue book),” 2018.

[112] M. Colesky, J.-H. Hoepman, and C. Hillen, “A critical analysis of privacy design strategies,” in 2016 IEEE Security and Privacy Workshops (SPW), 2016, pp. 33-40.

[113] L. Stark, J. King, X. Page, A. Lampinen, J. Vitak, P. Wisniewski, T. Whalen, and N. Good, “Bridging the gap between privacy by design and privacy in practice,” in Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, 2016, pp. 3415-3422.

[114] G.-F. Angelis, C. Timplalexis, S. Krinidis, D. Ioannidis, and D. Tzovaras, "NILM Applications: Literature review of learning approaches, recent developments and challenges," *Energy and Buildings*, pp. 111951, 2022.

[115] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE,* vol. 80, no. 12, pp. 1870-1891, 1992.

[116] C. Laughman, L. Kwangduk, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong, "Power signature analysis," *IEEE Power and Energy Magazine,* vol. 1, no. 2, pp. 56-63, 2003.

[117] P. Street, "Dataport: the world's largest energy data resource," *Pecan Street Inc*, 2015.

[118] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load Signature Study—Part I: Basic Concept, Structure, and Methodology," *IEEE Transactions on Power Delivery,* vol. 25, no. 2, pp. 551-560, 2010.

[119] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load Signature Study—Part II: Disaggregation Framework, Simulation, and Applications," *IEEE Transactions on Power Delivery,* vol. 25, no. 2, pp. 561-569, 2010.

[120] W. Kong, Z. Y. Dong, B. Wang, J. Zhao, and J. Huang, "A practical solution for non-intrusive type II load monitoring based on deep learning and post-processing," *IEEE Transactions on Smart Grid,* vol. 11, no. 1, pp. 148-160, 2019.

[121] E. Nashrullah, and A. Halim, "Performance Evaluation of Superstate HMM with Median Filter For Appliance Energy Disaggregation," in 2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), 2019, pp. 374-379.

[122] Y. Li, Z. Peng, J. Huang, Z. Zhang, and J. H. Son, "Energy disaggregation via hierarchical factorial hmm," in Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring, Austin, TX, USA, 2014.

[123] B. Zhao, K. He, L. Stankovic, and V. Stankovic, "Improving event-based non-intrusive load monitoring using graph signal processing," *IEEE Access,* vol. 6, pp. 53944-53959, 2018.

[124] C. Green, and S. Garimella, "Analysis of supervised graph signal processing-based load disaggregation for residential demand-side management," *Electric Power Systems Research,* vol. 208, pp. 107878, 2022.

[125] S. Verma, S. Singh, and A. Majumdar, "Multi-label LSTM autoencoder for non-intrusive appliance load monitoring," *Electric Power Systems Research,* vol. 199, pp. 107414, 2021.

[126] M. Kaselimi, N. Doulamis, A. Voulodimos, E. Protopapadakis, and A. Doulamis, "Context aware energy disaggregation using adaptive bidirectional LSTM models," *IEEE Transactions on Smart Grid,* vol. 11, no. 4, pp. 3054-3067, 2020.

[127] X. Zhou, J. Feng, and Y. Li, "Non-intrusive load decomposition based on CNN–LSTM hybrid deep learning model," *Energy Reports,* vol. 7, pp. 5762-5771, 2021.

[128] A. Yadav, A. Sinha, A. Saidi, C. Trinkl, and W. Zörner, "NILM based Energy Disaggregation Algorithm for Dairy Farms," in Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring, 2020, pp. 16-19.

[129] O. Krystalakos, C. Nalmpantis, and D. Vrakas, "Sliding window approach for online energy disaggregation using artificial neural networks," in Proceedings of the 10th Hellenic Conference on Artificial Intelligence, 2018, pp. 1-6.

[130] D. Ding, J. Li, K. Zhang, H. Wang, K. Wang, and T. Cao, "Non-intrusive load monitoring method with inception structured CNN," *Applied Intelligence*, pp. 1-18, 2021.

[131] D. Yang, X. Gao, L. Kong, Y. Pang, and B. Zhou, "An event-driven convolutional neural architecture for non-intrusive load monitoring of residential appliance," *IEEE Transactions on Consumer Electronics,* vol. 66, no. 2, pp. 173-182, 2020.

[132] F. Ciancetta, G. Bucci, E. Fiorucci, S. Mari, and A. Fioravanti, "A new convolutional neural network-based system for NILM applications," *IEEE Transactions on Instrumentation and Measurement,* vol. 70, pp. 1-12, 2020.

[133] M. M. R. Khan, M. A. B. Siddique, and S. Sakib, "Non-intrusive electrical appliances monitoring and classification using K-nearest neighbors," in 2019 2nd International Conference on Innovation in Engineering and Technology (ICIET), 2019, pp. 1-5.

[134] F. B. Gurbuz, R. Bayindir, and S. Vadi, "Comprehensive Non-Intrusive Load Monitoring Process: Device Event Detection, Device Feature Extraction and Device Identification Using KNN, Random Forest and Decision Tree," in 2021 10th International Conference on Renewable Energy Research and Application (ICRERA), 2021, pp. 447-452.

[135] S. Karnouskos, "Smart houses in the smart grid and the search for value-added services in the cloud of things era," in 2013 IEEE International Conference on Industrial Technology (ICIT), 2013, pp. 2016-2021.

[136] Y. Liu, X. Yang, W. Wen, and M. Xia, "Smarter Grid in the 5G Era: Integrating Power Internet of Things with Cyber Physical System," *Frontiers in Communications and Networks,* vol. 2, pp. 23, 2021.

[137] Y. Shahzad, H. Javed, H. Farman, J. Ahmad, B. Jan, and M. Zubair, "Internet of energy: Opportunities, applications, architectures and challenges in smart industries," *Computers & Electrical Engineering,* vol. 86, pp. 106739, 2020.

[138] M. Tao, J. Zuo, Z. Liu, A. Castiglione, and F. Palmieri, "Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes," *Future Generation Computer Systems,* vol. 78, pp. 1040-1051, 2018.

[139] J. Lloret, J. Tomas, A. Canovas, and L. Parra, "An integrated IoT architecture for smart metering," *IEEE Communications Magazine,* vol. 54, no. 12, pp. 50-57, 2016.

[140] A. Meloni, and L. Atzori, "A cloud-based and restful internet of things platform to foster smart grid technologies integration and re-usability," in 2016 IEEE International Conference on Communications Workshops (ICC), 2016, pp. 387-392.

[141] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in Proceedings of the 2015 workshop on mobile big data, 2015, pp. 37-42.

[142] Y. Yan, and W. Su, "A fog computing solution for advanced metering infrastructure," in 2016 IEEE/PES Transmission and Distribution Conference and Exposition (T&D), 2016, pp. 1-4.

[143] M. H. Y. Moghaddam, and A. Leon-Garcia, "A fog-based internet of energy architecture for transactive energy management systems," *IEEE Internet of Things Journal,* vol. 5, no. 2, pp. 1055-1069, 2018.

[144] M. Forcan, and M. Maksimović, "Cloud-fog-based approach for smart grid monitoring," *Simulation Modelling Practice and Theory,* vol. 101, pp. 101988, 2020.

[145] Y. H. Lin, "Novel smart home system architecture facilitated with distributed and embedded flexible edge analytics in demand‐side management," *International Transactions on Electrical Energy Systems,* vol. 29, no. 6, pp. e12014, 2019.

[146] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE network,* vol. 33, no. 2, pp. 111-117, 2019.

[147] M. Orlando, A. Estebsari, E. Pons, M. Pau, S. Quer, M. Poncino, L. Bottaccioli, and E. Patti, "A Smart Meter Infrastructure for Smart Grid IoT Applications," *IEEE Internet of Things Journal*, 2021.

[148] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine,* vol. 37, no. 3, pp. 50-60, 2020.

[149] X. Zhang, F. Fang, and J. Wang, "Probabilistic solar irradiation forecasting based on variational Bayesian inference with secure federated learning," *IEEE Transactions on Industrial Informatics*, 2020.

[150] Y. Wang, I. L. Bennani, X. Liu, M. Sun, and Y. Zhou, "Electricity Consumer Characteristics Identification: A Federated Learning Approach," *IEEE Transactions on Smart Grid*, 2021.

[151] S. Lee, and D.-H. Choi, "Federated reinforcement learning for energy management of multiple smart homes with distributed energy resources," *IEEE Transactions on Industrial Informatics*, 2020.

[152] S. A.-h. Soliman, and A. M. Al-Kandari, "8 - Dynamic Electric Load Forecasting," *Electrical Load Forecasting*, S. A.-h. Soliman and A. M. Al-Kandari, eds., pp. 291-352, Boston: Butterworth-Heinemann, 2010.

[153] S. N. Fallah, M. Ganjkhani, S. Shamshirband, and K.-w. Chau, "Computational intelligence on short-term load forecasting: A methodological overview," *Energies,* vol. 12, no. 3, pp. 393, 2019.

[154] C. Si, S. Xu, C. Wan, D. Chen, W. Cui, and J. Zhao, "Electric Load Clustering in Smart Grid: Methodologies, Applications, and Future Trends," *Journal of Modern Power Systems and Clean Energy,* vol. 9, no. 2, pp. 237-252, 2021.

[155] G. Dudek, "Pattern-based local linear regression models for short-term load forecasting," *Electric power systems research,* vol. 130, pp. 139-147, 2016.

[156] B. Dhaval, and A. Deshpande, "Short-term load forecasting with using multiple linear regression," *International Journal of Electrical and Computer Engineering,* vol. 10, no. 4, pp. 3911, 2020.

[157] M. Massaoudi, S. S. Refaat, I. Chihi, M. Trabelsi, F. S. Oueslati, and H. Abu-Rub, "A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for Short-Term Load Forecasting," *Energy,* vol. 214, pp. 118874, 2021.

[158] C.-M. Lee, and C.-N. Ko, "Short-term load forecasting using lifting scheme and ARIMA models," *Expert Systems with Applications,* vol. 38, no. 5, pp. 5902-5911, 2011.

[159] J. Wang, X. Chen, F. Zhang, F. Chen, and Y. Xin, "Building load forecasting using deep neural network with efficient feature fusion," *Journal of Modern Power Systems and Clean Energy,* vol. 9, no. 1, pp. 160-169, 2021.

[160] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Transactions on Smart Grid,* vol. 10, no. 1, pp. 841-851, 2017.

[161] S. Li, L. Goel, and P. Wang, "An ensemble approach for short-term load forecasting by extreme learning machine," *Applied Energy,* vol. 170, pp. 22-29, 2016.

[162] D. Gan, Y. Wang, S. Yang, and C. Kang, "Embedding based quantile regression neural network for probabilistic load forecasting," *Journal of Modern Power Systems and Clean Energy,* vol. 6, no. 2, pp. 244-254, 2018.

[163] Y. Wang, D. Gan, N. Zhang, L. Xie, and C. Kang, "Feature selection for probabilistic load forecasting via sparse penalized quantile regression," *Journal of Modern Power Systems and Clean Energy,* vol. 7, no. 5, pp. 1200-1209, 2019.

[164] H. Jahangir, H. Tayarani, S. S. Gougheri, M. A. Golkar, A. Ahmadian, and A. Elkamel, "Deep Learning-Based Forecasting Approach in Smart Grids With Microclustering and Bidirectional LSTM Network," *IEEE Transactions on Industrial Electronics,* vol. 68, no. 9, pp. 8298-8309, 2021.

[165] H. Jahangir, S. S. Gougheri, B. Vatandoust, M. A. Golkar, A. Ahmadian, and A. Hajizadeh, "Plug-in Electric Vehicle Behavior Modeling in Energy Market: A Novel Deep Learning-Based Approach With Clustering Technique," *IEEE Transactions on Smart Grid,* vol. 11, no. 6, pp. 4738-4748, 2020.

[166] M. Sun, Y. Wang, F. Teng, Y. Ye, G. Strbac, and C. Kang, "Clustering-based residential baseline estimation: A probabilistic perspective," *IEEE Transactions on Smart Grid,* vol. 10, no. 6, pp. 6014-6028, 2019.

[167] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," *IEEE Transactions on Smart Grid,* vol. 6, no. 2, pp. 911-918, 2014.

[168] Y. Li, D. Han, and Z. Yan, "Long-term system load forecasting based on data-driven linear clustering method," *Journal of Modern Power Systems and Clean Energy,* vol. 6, no. 2, pp. 306-316, 2018.

[169] J. Mathew, and R. K. Behera, "EMD-Att-LSTM: A Data-Driven Strategy Combined with Deep Learning for Short-Term Load Forecasting," *Journal of Modern Power Systems and Clean Energy*, 2021.

[170] X. Shi, X. Lei, Q. Huang, S. Huang, K. Ren, and Y. Hu, "Hourly day-ahead wind power prediction using the hybrid model of variational model decomposition and long short-term memory," *Energies,* vol. 11, no. 11, pp. 3227, 2018.

[171] S. Kim, G. Lee, G.-Y. Kwon, D.-I. Kim, and y.-j. Shin, "Deep Learning Based on Multi-Decomposition for Short-Term Load Forecasting," *Energies,* vol. 11, pp. 3433, 12/07, 2018.

[172] R. J. Hyndman, and G. Athanasopoulos, *Forecasting: principles and practice*: OTexts, 2018.

[173] H. Liu, and Z. Long, "An improved deep learning model for predicting stock market price time series," *Digital Signal Processing,* vol. 102, pp. 102741, 2020.

[174] Z. Zhu, Y. Sun, and H. Li, "Hybrid of EMD and SVMs for short-term load forecasting," in 2007 IEEE International Conference on Control and Automation, 2007, pp. 1044-1047.

[175] Y. K. Semero, J. Zhang, and D. Zheng, "EMD–PSO–ANFIS-based hybrid approach for short-term load forecasting in microgrids," *IET Generation, Transmission Distribution,* vol. 14, no. 3, pp. 470-475, 2019.

[176] K. Dragomiretskiy, and D. Zosso, "Variational mode decomposition," *IEEE transactions on signal processing,* vol. 62, no. 3, pp. 531-544, 2013.

[177] B. C. Erdener, C. Feng, K. Doubleday, A. Florita, and B.-M. Hodge, "A review of behind-the-meter solar forecasting," *Renewable and Sustainable Energy Reviews,* vol. 160, pp. 112224, 2022.

[178] M. Tabone, S. Kiliccote, and E. C. Kara, "Disaggregating solar generation behind individual meters in real time," in Proceedings of the 5th Conference on Systems for Built Environments, 2018, pp. 43-52.

[179] J. Brown, A. Abate, and A. Rogers, "Disaggregation of household solar energy generation using censored smart meter data," *Energy and Buildings,* vol. 231, pp. 110617, 2021.

[180] C. M. Cheung, W. Zhong, C. Xiong, A. Srivastava, R. Kannan, and V. K. Prasanna, "Behind-the-meter solar generation disaggregation using consumer mixture models," in 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2018, pp. 1-6.

[181] E. C. Kara, C. M. Roberts, M. Tabone, L. Alvarez, D. S. Callaway, and E. M. Stewart, "Disaggregating solar generation from feeder-level measurements," *Sustainable Energy, Grids Networks,* vol. 13, pp. 112-121, 2018.

[182] F. Sossan, L. Nespoli, V. Medici, and M. Paolone, "Unsupervised Disaggregation of PhotoVoltaic Production from Composite Power Flow Measurements of Heterogeneous Prosumers," *IEEE Transactions on Industrial Informatics,* vol. PP, 06/15, 2017.

[183] K. Li, F. Wang, Z. Mi, M. Fotuhi-Firuzabad, N. Duić, and T. Wang, "Capacity and output power estimation approach of individual behind-the-meter distributed photovoltaic system for demand response baseline estimation," *Applied Energy,* vol. 253, pp. 113595, 2019/11/01/, 2019.

[184] M. Wytock, and J. Kolter, "Contextually supervised source separation with application to energy disaggregation," in Proceedings of the AAAI Conference on Artificial Intelligence, 2014.

[185] E. Vrettos, E. C. Kara, E. M. Stewart, and C. Roberts, "Estimating PV power from aggregate power measurements within the distribution grid," *Journal of Renewable and Sustainable Energy* vol. 11, no. 2, pp. 023707, 2019.

[186] H. Shaker, H. Zareipour, and D. Wood, "A data-driven approach for estimating the power generation of invisible solar sites," *IEEE Transactions on Smart Grid,* vol. 7, no. 5, pp. 2466-2476, 2015.

[187] H. Shaker, H. Zareipour, and D. Wood, "Estimating power generation of invisible solar sites using publicly available data," *IEEE Transactions on Smart Grid,* vol. 7, no. 5, pp. 2456-2465, 2016.

[188] J. Bright, S. Killinger, D. Lingfors, and N. Engerer, "Improved satellite-derived PV power nowcasting using real-time power data from reference PV systems," *Solar Energy,* vol. 168, pp. 118-139, 07/01, 2018.

[189] M. Pierro, M. De Felice, E. Maggioni, D. Moser, A. Perotto, F. Spada, and C. Cornaro, "Data-driven upscaling methods for regional photovoltaic power estimation and forecast using satellite and numerical weather prediction data," *Solar Energy,* vol. 158, pp. 1026-1038, 2017.

[190] F. Bu, K. Dehghanpour, Y. Yuan, Z. Wang, and Y. Zhang, "A Data-Driven Game-Theoretic Approach for Behind-the-Meter PV Generation Disaggregation," *IEEE Transactions on Power Systems,* vol. PP, no. 99, pp. 1-1, 2020.

[191] Y. Wang, N. Zhang, Q. Chen, D. S. Kirschen, P. Li, and Q. Xia, "Data-Driven Probabilistic Net Load Forecasting With High Penetration of Behind-the-Meter PV," *IEEE Transactions on Power Systems,* vol. 33, no. 3, pp. 3255-3264, 2018.

[192] X. Y. Zhang, S. Kuenzel, and C. Watkins, "Feeder-Level Deep Learning-based Photovoltaic Penetration Estimation Scheme," in 2020 12th IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), 2020, pp. 1-5.

[193] N. Oceanic, and A. Administration, "National Centers for Environmental Information (NCEI)," *Historical Palmer Drought Indices.*, 2016.

[194] S. Burt, "The Climatological Observers Link (COL) at 50," *Weather,* vol. 75, no. 5, pp. 137-144, 2020.

[195] M. Sengupta, Y. Xie, A. Lopez, A. Habte, G. Maclaurin, and J. Shelby, "The national solar radiation data base (NSRDB)," *Renewable and sustainable energy reviews,* vol. 89, pp. 51-60, 2018.

[196] S. Cox, N. Lee, and M. Jacobson, *RE Data Explorer: Use of the Tool to Support Renewable Energy Project Development*, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2020.

[197] Y. Wen, D. AlHakeem, P. Mandal, S. Chakraborty, Y.-K. Wu, T. Senjyu, S. Paudyal, and T.-L. Tseng, "Performance evaluation of probabilistic methods based on bootstrap and quantile regression to quantify PV power point forecast uncertainty," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 31, no. 4, pp. 1134-1144, 2019.

[198] D. W. Van der Meer, J. Munkhammar, and J. Widén, "Probabilistic forecasting of solar power, electricity consumption and net load: Investigating the effect of seasons, aggregation and penetration on prediction intervals," *Solar Energy,* vol. 171, pp. 397-413, 2018.

[199] A. Ahmed Mohammed, and Z. Aung, "Ensemble learning approach for probabilistic forecasting of solar power generation," *Energies,* vol. 9, no. 12, pp. 1017, 2016.

[200] M. Sun, T. Zhang, Y. Wang, G. Strbac, and C. Kang, "Using Bayesian deep learning to capture uncertainty for residential net load forecasting," *IEEE Transactions on Power Systems,* vol. 35, no. 1, pp. 188-201, 2019.

[201] L. Waswa, M. J. Chihota, and B. Bekker, "A Probabilistic Estimation of PV Capacity in Distribution Networks From Aggregated Net-Load Data," *IEEE Access,* vol. 9, pp. 140358-140371, 2021.

[202] J. Lin, J. Ma, and J. Zhu, "A Privacy-Preserving Federated Learning Method for Probabilistic Community-Level Behind-the-Meter Solar Generation

Disaggregation," *IEEE Transactions on Smart Grid,* vol. 13, no. 1, pp. 268-279, 2021.

[203] X. Zhang, F. Fang, and J. Wang, "Probabilistic Solar Irradiation Forecasting Based on Variational Bayesian Inference With Secure Federated Learning," *IEEE Transactions on Industrial Informatics,* vol. 17, no. 11, pp. 7849-7859, 2021.

[204] W. Li, M. Yi, M. Wang, Y. Wang, D. Shi, and Z. Wang, "Real-Time Energy Disaggregation at Substations With Behind-the-Meter Solar Generation," *IEEE Transactions on Power Systems,* vol. 36, no. 3, pp. 2023-2034, 2021.

[205] G. S. Ledva, L. Balzano, and J. L. Mathieu, "Real-time energy disaggregation of a distribution feeder's demand using online learning," *IEEE Transactions on Power Systems,* vol. 33, no. 5, pp. 4730-4740, 2018.

[206] G. S. Ledva, and J. L. Mathieu, "Separating feeder demand into components using substation, feeder, and smart meter measurements," *IEEE Transactions on Smart Grid,* vol. 11, no. 4, pp. 3280-3290, 2020.

[207] J. Wang, X. Zhu, M. Liang, Y. Meng, A. Kling, D. L. Lubkeman, and N. Lu, "A Data-Driven Pivot-Point-Based Time-Series Feeder Load Disaggregation Method," *IEEE Transactions on Smart Grid,* vol. 11, no. 6, pp. 5396-5406, 2020.

[208] S. Wang, R. Li, A. Evans, and F. Li, "Regional nonintrusive load monitoring for low voltage substations and distributed energy resources," *Applied Energy,* vol. 260, pp. 114225, 2020.

[209] M. Cui, M. Khodayar, C. Chen, X. Wang, Y. Zhang, and M. E. Khodayar, "Deep Learning-Based Time-Varying Parameter Identification for System-Wide Load Modeling," *IEEE Transactions on Smart Grid,* vol. 10, no. 6, pp. 6102-6114, 2019.

[210] Y. Xu, and J. Milanovic, "Artificial-Intelligence-Based Methodology for Load Disaggregation at Bulk Supply Point," *IEEE Transactions on Power Systems,* vol. 30, no. 2, pp. 795-803, 2014.

[211] M. W. Asres, A. A. Girmay, C. Camarda, and G. T. Tesfamariam, "Non-intrusive load composition estimation from aggregate ZIP load models using machine learning," *International Journal of Electrical Power & Energy Systems,* vol. 105, pp. 191-200, 2019.

[212] F. Bu, K. Dehghanpour, Y. Yuan, Z. Wang, and Y. Guo, "Disaggregating Customer-level Behind-the-Meter PV Generation Using Smart Meter Data and Solar Exemplars," *IEEE Transactions on Power Systems*, pp. 1-1, 2021.

[213] F. He, J. Zhou, L. Mo, K. Feng, G. Liu, and Z. He, "Day-ahead short-term load probability density forecasting method with a decomposition-based quantile regression forest," *Applied Energy,* vol. 262, pp. 114396, 2020.

[214] S. Zhang, Y. Wang, Y. Zhang, D. Wang, and N. Zhang, "Load probability density forecasting by transforming and combining quantile forecasts," *Applied Energy,* vol. 277, pp. 115600, 2020/11/01/, 2020.

[215] W. Zhang, H. Quan, and D. Srinivasan, "Parallel and reliable probabilistic load forecasting via quantile regression forest and quantile determination," *Energy,* vol. 160, pp. 810-819, 2018.

[216] Y. Wang, D. Gan, M. Sun, N. Zhang, Z. Lu, and C. Kang, "Probabilistic individual load forecasting using pinball loss guided LSTM," *Applied Energy,* vol. 235, pp. 10-20, 2019.

[217] J. Xie, and T. Hong, "Variable selection methods for probabilistic load forecasting: Empirical evidence from seven states of the united states," *IEEE Transactions on Smart Grid,* vol. 9, no. 6, pp. 6039-6046, 2017.

[218] J. F. Toubeau, T. Morstyn, J. Bottieau, K. Zheng, D. Apostolopoulou, Z. D. Grève, Y. Wang, and F. Vallée, "Capturing Spatio-Temporal Dependencies in the Probabilistic Forecasting of Distribution Locational Marginal Prices," *IEEE Transactions on Smart Grid,* vol. 12, no. 3, pp. 2663-2674, 2021.

[219] F. Fan, K. Bell, and D. Infield, "Probabilistic Real-Time Thermal Rating Forecasting for Overhead Lines by Conditionally Heteroscedastic Auto-Regressive Models," *IEEE Transactions on Power Delivery,* vol. 32, no. 4, pp. 1881-1890, 2017.

[220] F. Fan, K. Bell, and D. Infield, "Transient-state real-time thermal rating forecasting for overhead lines by an enhanced analytical method," *Electric Power Systems Research,* vol. 167, pp. 213-221, 2019/02/01/, 2019.

[221] Y. He, and H. Li, "Probability density forecasting of wind power using quantile regression neural network and kernel density estimation," *Energy conversion and management,* vol. 164, pp. 374-384, 2018.

[222] M. Khashei, and M. Bijari, "Hybridization of the probabilistic neural networks with feed-forward neural networks for forecasting," *Engineering Applications of Artificial Intelligence,* vol. 25, no. 6, pp. 1277-1288, 2012/09/01/, 2012.

[223] R. M. Pattanayak, H. S. Behera, and S. Panigrahi, "A novel probabilistic intuitionistic fuzzy set based model for high order fuzzy time series forecasting," *Engineering Applications of Artificial Intelligence,* vol. 99, pp. 104136, 2021.

[224] J. Ponoćko, and J. V. Milanović, "Forecasting Demand Flexibility of Aggregated Residential Load Using Smart Meter Data," *IEEE Transactions on Power Systems,* vol. 33, no. 5, pp. 5446-5455, 2018.

[225] L. Li, K. Ota, and M. Dong, "When weather matters: IoT-based electrical load forecasting for smart grid," *IEEE Communications Magazine,* vol. 55, no. 10, pp. 46-51, 2017.

[226] L. Wen, K. Zhou, S. Yang, and L. Li, "Compression of smart meter big data: A survey," *Renewable and Sustainable Energy Reviews,* vol. 91, pp. 59-69, 2018/08/01/, 2018.

[227] Y. Wang, Q. Chen, T. Hong, and C. Kang, "Review of smart meter data analytics: Applications, methodologies, and challenges," *IEEE Transactions on Smart Grid,* vol. 10, no. 3, pp. 3125-3148, 2018.

[228] Z. A. Khan, D. Jayaweera, and M. S. Alvarez-Alvarado, “A novel approach for load profiling in smart power grids using smart meter data,” *Electric Power Systems Research,* vol. 165, pp. 191-198, 2018.

[229] W. Li, T. Logenthiran, V.-T. Phan, and W. L. Woo, “A novel smart energy theft system (SETS) for IoT-based smart home,” *IEEE Internet of Things Journal,* vol. 6, no. 3, pp. 5531-5539, 2019.

[230] Y. Wang, M. Jia, N. Gao, L. V. Krannichfeldt, M. Sun, and G. Hug, “Federated Clustering for Electricity Consumption Pattern Extraction,” *IEEE Transactions on Smart Grid*, pp. 1-1, 2022.

[231] B. Wang, Y. Li, W. Ming, and S. Wang, “Deep reinforcement learning method for demand response management of interruptible load,” *IEEE Transactions on Smart Grid,* vol. 11, no. 4, pp. 3146-3155, 2020.

[232] L. Wen, K. Zhou, J. Li, and S. Wang, “Modified deep learning and reinforcement learning for an incentive-based demand response model,” *Energy,* vol. 205, pp. 118019, 2020.

[233] Y. Zhang, T. Huang, and E. F. Bompard, “Big data analytics in smart grids: a review,” *Energy Informatics,* vol. 1, no. 1, pp. 8, 2018/08/13, 2018.

[234] R. Ma, H.-H. Chen, Y.-R. Huang, and W. Meng, “Smart grid communication: Its challenges and opportunities,” *IEEE transactions on Smart Grid,* vol. 4, no. 1, pp. 36-46, 2013.

[235] M. Shateri, F. Messina, P. Piantanida, and F. Labeau, “Real-time privacy-preserving data release for smart meters,” *IEEE Transactions on Smart Grid,* vol. 11, no. 6, pp. 5174-5183, 2020.

[236] M. Shateri, F. Messina, P. Piantanida, and F. Labeau, “Learning Sparse Privacy-Preserving Representations for Smart Meters Data,” in 2021 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2021, pp. 333-338.

[237] S. Ramírez-Gallego, B. Krawczyk, S. García, M. Woźniak, and F. Herrera, “A survey on data preprocessing for data stream mining: Current status and future directions,” *Neurocomputing,* vol. 239, pp. 39-57, 2017.

[238] S. García, S. Ramírez-Gallego, J. Luengo, J. M. Benítez, and F. Herrera, “Big data preprocessing: methods and prospects,” *Big Data Analytics,* vol. 1, no. 1, pp. 1-22, 2016.

[239] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, “Data preprocessing for supervised leaning,” *International journal of computer science,* vol. 1, no. 2, pp. 111-117, 2006.

[240] C. Seger, "An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing," 2018.

[241] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*: MIT press, 2016.

[242] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature,* vol. 521, no. 7553, pp. 436-444, 2015.

[243] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*: O'Reilly Media, 2019.

[244] C. R. Taylor, "Dynamic programming and the curses of dimensionality," *Applications of dynamic programming to agricultural decision problems*, pp. 1-10: CRC Press, 2019.

[245] G. Chandrashekar, and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering,* vol. 40, no. 1, pp. 16-28, 2014.

[246] T. M. Cover, and J. A. Thomas, *Elements of information theory*: John Wiley & Sons, 2012.

[247] S. Russell, and P. Norvig, "Artificial intelligence: a modern approach," 2002.

[248] A. L. Samuel, "Artificial intelligence: a frontier of automation," *The Annals of the American Academy of Political and Social Science,* vol. 340, no. 1, pp. 10-20, 1962.

[249] F. Chollet, *Deep learning with Python*: Simon and Schuster, 2021.

[250] K. Parrish, "Deep learning vs. machine learning: what's the difference between the two," *Online: https://www. digitaltrends. com/cool-tech/deep-learning-vs-machine-learning-explained/2/[Last accessed: 08.05. 2018]*, 2018.

[251] R. Caruana, and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in Proceedings of the 23rd international conference on Machine learning, 2006, pp. 161-168.

[252] M. Mohri, A. Rostamizadeh, and A. Talwalkar, "Foundations of machine learning.[Sl]," The MIT Press, 2012.

[253] M. Belgiu, and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS journal of photogrammetry and remote sensing,* vol. 114, pp. 24-31, 2016.

[254] S. R. Safavian, and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE transactions on systems, man, and cybernetics,* vol. 21, no. 3, pp. 660-674, 1991.

[255] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?," in International workshop on machine learning and data mining in pattern recognition, 2012, pp. 154-168.

[256] A. Natekin, and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics,* vol. 7, no. 21, 2013-December-04, 2013.

[257] T. Chen, "Introduction to boosted trees," *University of Washington Computer Science,* vol. 22, no. 2014, pp. 115, 2014.

[258] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems,* vol. 30, pp. 3146-3154, 2017.

[259] H. B. Barlow, “Unsupervised learning,” *Neural computation,* vol. 1, no. 3, pp. 295-311, 1989.

[260] A. Kulkarni, D. Chong, and F. A. Batarseh, "5 - Foundations of data imbalance and solutions for a data democracy," *Data Democracy*, F. A. Batarseh and R. Yang, eds., pp. 83-106: Academic Press, 2020.

[261] S.-C. Wang, "Artificial neural network," *Interdisciplinary computing in java programming*, pp. 81-100: Springer, 2003.

[262] G. E. Hinton, “Deep belief networks,” *Scholarpedia,* vol. 4, no. 5, pp. 5947, 2009.

[263] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” *IEEE transactions on neural networks and learning systems,* vol. 32, no. 1, pp. 4-24, 2020.

[264] D. W. Ruck, S. K. Rogers, and M. Kabrisky, “Feature selection using a multilayer perceptron,” *Journal of Neural Network Computing,* vol. 2, no. 2, pp. 40-48, 1990.

[265] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in 2017 International Conference on Engineering and Technology (ICET), 2017, pp. 1-6.

[266] S. Herculano-Houzel, “Numbers of neurons as biological correlates of cognitive capability,” *Current Opinion in Behavioral Sciences,* vol. 16, pp. 1-7, 2017.

[267] R. A. Alzahrani, and A. C. Parker, “Neuromorphic circuits with neural modulation enhancing the information content of neural signaling,” in International Conference on Neuromorphic Systems 2020, 2020, pp. 1-8.

[268] K. Gurney, *An introduction to neural networks*: CRC press, 2018.

[269] G. Zhang, C. Wang, B. Xu, and R. Grosse, “Three mechanisms of weight decay regularization,” *arXiv preprint arXiv:1810.12281*, 2018.

[270] P. Baldi, and P. J. Sadowski, “Understanding dropout,” *Advances in neural information processing systems,* vol. 26, pp. 2814-2822, 2013.

[271] C. Hansen, “Optimizers explained-adam, momentum and stochastic gradient descent,” *Consulté le,* vol. 12, no. 04, 2020.

[272] A. Kathuria, "Intro to optimization in deep learning: Momentum, rmsprop and adam," 2018.

[273] N. Ketkar, “Convolutional Neural Networks,” *Springer International Publishing*, 2017.

[274] P. Kim, "Convolutional neural network," *MATLAB deep learning*, pp. 121-147: Springer, 2017.

[275] J. Nagi, F. Ducatelle, G. A. Di Caro, D. Cireşan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella, “Max-pooling convolutional neural networks for vision-based hand gesture recognition,” in 2011 IEEE

International Conference on Signal and Image Processing Applications (ICSIPA), 2011, pp. 342-347.

[276] L. R. Medsker, and L. Jain, “Recurrent neural networks,” *Design and Applications,* vol. 5, 2001.

[277] L. R. Medsker, and L. Jain, “Recurrent neural networks,” *Design and Applications,* vol. 5, pp. 64-67, 2001.

[278] S. Hochreiter, and J. Schmidhuber, “Long short-term memory,” *Neural computation,* vol. 9, no. 8, pp. 1735-1780, 1997.

[279] R. Dey, and F. M. Salem, “Gate-variants of gated recurrent unit (GRU) neural networks,” in 2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS), 2017, pp. 1597-1600.

[280] R. Pascanu, T. Mikolov, and Y. Bengio, “On the difficulty of training recurrent neural networks,” in International conference on machine learning, 2013, pp. 1310-1318.

[281] P. Mack, "Chapter 35 - Big Data, Data Mining, and Predictive Analytics and High Performance Computing," *Renewable Energy Integration*, L. E. Jones, ed., pp. 439-454, Boston: Academic Press, 2014.

[282] M. Manbachi, "Chapter 5 - Impact of Distributed Energy Resource Penetrations on Smart Grid Adaptive Energy Conservation and Optimization Solutions," *Operation of Distributed Energy Resources in Smart Distribution Networks*, K. Zare and S. Nojavan, eds., pp. 101-138: Academic Press, 2018.

[283] H. Wang, J. Zhang, C. Lu, and C. Wu, “Privacy Preserving in Non-Intrusive Load Monitoring: A Differential Privacy Perspective,” *IEEE Transactions on Smart Grid*, 2020.

[284] N. Grid, T. Smart, G. Interoperability, C. Security, and W. Group, *Introduction to NISTIR 7628 Guidelines for Smart Grid Cyber Security*: Introduction to NISTIR 7628 Guidelines for Smart Grid Cyber Security, 2010.

[285] L. Sandys, *Energy Data Taskforce report: A strategy for a Modern Digitalised Energy System*, 2019.

[286] A. Alabdulkarim, Z. Lukszo, and T. W. Fens, “Acceptance of Privacy-Sensitive Infrastructure Systems: A Case of Smart Metering in The Netherlands,” 2012.

[287] C. Cuijpers, and B. J. Koops, “Smart Metering and Privacy in Europe: Lessons from the Dutch Case,” *Social Science Electronic Publishing*, 2013.

[288] S. Li, A. Khisti, and A. Mahajan, “Privacy-optimal strategies for smart metering systems with a rechargeable battery,” in 2016 American Control Conference (ACC), 2016, pp. 2080-2085.

[289] P. Barbosa, A. Brito, and H. Almeida, “A Technique to provide differential privacy for appliance usage in smart metering,” *Information Sciences,* vol. 370-371, pp. 355-367, 2016/11/20/, 2016.

[290] L. Zhang, and J. Zhang, "Publicly Verifiable Spatial and Temporal Aggregation Scheme Against Malicious Aggregator in Smart Grid," *Applied Sciences,* vol. 9, no. 3, pp. 490, 2019.

[291] U. Nations, "Adoption of the Paris agreement," 2015.

[292] E. Commission, "Directive 2012/27/EU of 25 October 2012 on energy efficiency, amending Directives 2009/125/EC and 2010/30/EU and repealing Directives 2004/8/EC and 2006/32/EC," 2012.

[293] G. Danezis, C. Fournet, M. Kohlweiss, and S. Zanella-Béguelin, "Smart meter aggregation via secret-sharing," in Proceedings of the first ACM workshop on Smart energy grid security, 2013, pp. 75-80.

[294] L. Brandeis, and S. Warren, "The right to privacy," *Harvard law review,* vol. 4, no. 5, pp. 193-220, 1890.

[295] R. Clarke, "What's 'Privacy'?," 2003.

[296] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in Theory of cryptography conference, 2006, pp. 265-284.

[297] F. Raji, A. Miri, and M. Jazi, "CP2: Cryptographic privacy protection framework for online social networks," *Computers & Electrical Engineering,* vol. 39, no. 7, pp. 2282-2298, 2013.

[298] S. Eskandarian, H. Corrigan-Gibbs, M. Zaharia, and D. Boneh, "Express: Lowering the Cost of Metadata-hiding Communication with Cryptographic Privacy," 2019.

[299] P. Derbeko, S. Dolev, E. Gudes, and S. Sharma, "Security and privacy aspects in MapReduce on clouds: A survey," *Computer Science Review,* vol. 20, pp. 1-28, 2016/05/01/, 2016.

[300] A. Paverd, A. Martin, and I. Brown, *Modelling and Automatically Analysing Privacy Properties for Honest-but-Curious Adversaries*, 2014.

[301] P. Voigt, and A. Von dem Bussche, *The eu general data protection regulation (gdpr): A Practical Guide*

Springer International Publishing, 2017.

[302] D. P. Act, "Data protection act," *London Station Off,* vol. 5, 2018.

[303] E. Directive, "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data," *Official Journal of the European Communities,* vol. 38, no. 281, pp. 31-50, 1995.

[304] Z. Erkin, J. R. Troncoso-Pastoriza, R. L. Lagendijk, and F. Pérez-González, "Privacy-preserving data aggregation in smart metering systems: An overview," *IEEE Signal Processing Magazine,* vol. 30, no. 2, pp. 75-86, 2013.

[305] M. A. Lisovich, D. K. Mulligan, and S. B. Wicker, "Inferring personal information from demand-response systems," *IEEE Security & Privacy,* vol. 8, no. 1, pp. 11-20, 2010.

[306] J. Durkay, D. Freeman, and K. Brangoccio, "Electricity Use in Marijuana Production," in National Conference of State Legislatures, 2016.

[307] P. D. DeVries, "An analysis of cryptocurrency, bitcoin, and the future," *International Journal of Business Management and Commerce,* vol. 1, no. 2, pp. 1-9, 2016.

[308] T. Hargreaves, M. Nye, and J. Burgess, "Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors," *Energy Policy,* vol. 38, no. 10, pp. 6111-6119, 2010/10/01/, 2010.

[309] Supervisor, and A. Berentsen, "Bitcoin Mining: An Economic Analysis."

[310] H. Bao, and R. Lu, "A new differentially private data aggregation with fault tolerance for smart grid communications," *IEEE Internet of Things Journal,* vol. 2, no. 3, pp. 248-258, 2015.

[311] L. Zhu, M. Li, Z. Zhang, X. Du, and M. Guizani, "Big data mining of users' energy consumption patterns in the wireless smart grid," *IEEE Wireless Communications,* vol. 25, no. 1, pp. 84-89, 2018.

[312] H. Li, L. Lai, and W. Zhang, "Communication requirement for reliable and secure state estimation and control in smart grid," *IEEE Transactions on Smart Grid,* vol. 2, no. 3, pp. 476-486, 2011.

[313] E. COMMISSION, *A joint contribution of DG ENER and DG INFSO towards the Digital Agenda, Action 73: Set of common functional requirements of the SMART METER*, 2011.

[314] W. W. Hogan, "Fairness and Dynamic Pricing: Comments," *The Electricity Journal,* vol. 23, no. 6, pp. 28-35, 2010/07/01/, 2010.

[315] X. Kong, C. Li, F. Zheng, and C. Wang, "Improved deep belief network for short-term load forecasting considering demand-side management," *IEEE Transactions on Power Systems,* vol. 35, no. 2, pp. 1531-1538, 2019.

[316] J. M. Bright, S. Killinger, D. Lingfors, and N. A. J. S. E. Engerer, "Improved satellite-derived PV power nowcasting using real-time power data from reference PV systems," vol. 168, pp. 118-139, 2018.

[317] E. Vrettos, E. C. Kara, E. M. Stewart, and C. Roberts, "Estimating PV power from aggregate power measurements within the distribution grid," *Journal of Renewable and Sustainable Energy,* vol. 11, no. 2, 2019.

[318] G. S. Ledva, L. Balzano, and J. L. J. I. T. o. P. S. Mathieu, "Real-time energy disaggregation of a distribution feeder's demand using online learning," vol. 33, no. 5, pp. 4730-4740, 2018.

[319] A. Pitì, G. Verticale, C. Rottondi, A. Capone, and L. Lo Schiavo, "The Role of Smart Meters in Enabling Real-Time Energy Services for Households: The Italian Case," *Energies,* vol. 10, pp. 199, 02/10, 2017.

[320] J. Torriti, *Peak energy demand and demand side response*: Routledge, 2015.

[321] M. Qadrdan, M. Cheng, J. Wu, and N. Jenkins, "Benefits of demand-side response in combined gas and electricity networks," *Applied Energy,* vol. 192, pp. 360-369, 2017.

[322] J.-R. Córdoba-Pachón, *Managing creativity: A systems thinking journey*: Routledge, 2018.

[323] M. Kornatka, and T. Popławski, "Advanced Metering Infrastructure—Towards a Reliable Network," *Energies,* vol. 14, no. 18, pp. 5986, 2021.

[324] P. G. C.R, A. Ramesh, D. Satvik, S. Nagasundari, and P. B. Honnavalli, "Simulation of SCADA System for Advanced Metering Infrastructure in Smart Grid." pp. 1071-1077.

[325] T. Ananthapadmanabha, A. Kulkarni, and A. Suma, "Automatic meter reading (AMR) based distribution security monitoring and distribution-supervisory control and data acquisition (D-SCADA) control," *Journal of Electrical and Electronics Engineering Research,* vol. 3, no. 6, pp. 108-120, 2011.

[326] K. P. Schneider, Y. Chen, D. P. Chassin, R. G. Pratt, D. W. Engel, and S. E. Thompson, *Modern grid initiative distribution taxonomy final report*, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2008.

[327] O. Parson, G. Fisher, A. Hersey, N. Batra, J. Kelly, A. Singh, W. Knottenbelt, and A. Rogers, "Dataport and NILMTK: A building data set designed for non-intrusive load monitoring," in 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), 2015, pp. 210-214.

[328] "Essential regulatory requirements and recommendations for data handling, data safety, and consumer protection," Technical Report version 1.0, 2011. https://ec. europa. eu/energy/sites/ener ….

[329] V. Tudor, M. Almgren, and M. Papatriantafilou, "The influence of dataset characteristics on privacy preserving methods in the advanced metering infrastructure," *Computers & Security,* vol. 76, pp. 178-196, 2018.

[330] J. Kelly, and W. Knottenbelt, *Neural NILM: Deep Neural Networks Applied to Energy Disaggregation*, 2015.

[331] J. Kolter, and M. Johnson, "REDD: A Public Data Set for Energy Disaggregation Research," *Artif. Intell.,* vol. 25, 01/01, 2011.

[332] C. E. Kement, H. Gultekin, B. Tavli, T. Girici, and S. Uludag, "Comparative analysis of load-shaping-based privacy preservation strategies in a smart grid," *IEEE Transactions on Industrial Informatics,* vol. 13, no. 6, pp. 3226-3235, 2017.

[333] F. Saghezchi, F. Saghezchi, A. Nascimento, and J. Rodriguez, "Game Theory and Pricing Strategies for Demand-Side Management in the Smart Grid," in 2014 9th International Symposium on Communication Systems, Networks and Digital Signal Processing, CSNDSP 2014, 2014.

[334] G. Eibl, and D. Engel, “Influence of data granularity on smart meter privacy,” *IEEE Transactions on Smart Grid,* vol. 6, no. 2, pp. 930-939, 2014.

[335] N. Buescher, S. Boukoros, S. Bauregger, and S. Katzenbeisser, “Two is not enough: Privacy assessment of aggregation schemes in smart metering,” *Proceedings on Privacy Enhancing Technologies,* vol. 4, no. 2017, pp. 198-214, 2017.

[336] P. Delforge, L. Schmidt, and S. Schmidt, "Home Idle Load: Devices Wasting Huge Amounts of Electricity When Not in Active Use(Tech.). NRDC," 2015.

[337] J. Kim, and H. Kim, “Classification performance using gated recurrent unit recurrent neural network on energy disaggregation,” in 2016 international conference on machine learning and cybernetics (ICMLC), 2016, pp. 105-110.

[338] J. Kelly, and W. Knottenbelt, “Neural nilm: Deep neural networks applied to energy disaggregation,” in Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments, 2015, pp. 55-64.

[339] M. DrIncecco, S. Squartini, and M. Zhong, “Transfer Learning for Non-Intrusive Load Monitoring,” *IEEE Transactions on Smart Grid,* vol. 11, no. 2, pp. 1419-1429, 2019.

[340] S. Kang, and J. W. Yoon, “Classification of home appliance by using Probabilistic KNN with sensor data,” in 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2016, pp. 1-5.

[341] F. Hidiyanto, and A. Halim, “KNN Methods with Varied K, Distance and Training Data to Disaggregate NILM with Similar Load Characteristic,” in Proceedings of the 3rd Asia Pacific Conference on Research in Industrial and Systems Engineering 2020, 2020, pp. 93-99.

[342] V. T. Hayashi, R. Arakaki, T. Y. Fujii, K. A. Khalil, and F. H. Hayashi, “B2B B2C Architecture for Smart Meters using IoT and Machine Learning: a Brazilian Case Study,” in 2020 International Conference on Smart Grids and Energy Systems (SGES), 2020, pp. 826-831.

[343] Y. Gao, Z. Pan, H. Wang, and G. Chen, “Alexa, my love: Analyzing reviews of amazon echo,” in 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2018, pp. 372-380.

[344] K. Noda, “Google Home: smart speaker as environmental control unit,” *Disability and rehabilitation: assistive technology,* vol. 13, no. 7, pp. 674-675, 2018.

[345] X. Y. Zhang, C. Watkins, C. C. Took, and S. Kuenzel, “Privacy boundary determination of smart meter data using an artificial intelligence adversary,” *International Transactions on Electrical Energy Systems*, 2021.

[346] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials,* vol. 22, no. 3, pp. 2031-2063, 2020.

[347] S. Zhang, J. Rong, and B. Wang, "A privacy protection scheme of smart meter for decentralized smart home environment based on consortium blockchain," *International Journal of Electrical Power & Energy Systems,* vol. 121, pp. 106140, 2020.

[348] J. Liu, and X. Gong, "Attention mechanism enhanced LSTM with residual architecture and its application for protein-protein interaction residue pairs prediction," *BMC Bioinformatics,* vol. 20, no. 1, pp. 609, 2019/11/27, 2019.

[349] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[350] C. Raffel, and D. P. Ellis, "Feed-forward networks with attention can solve some long-term memory problems," *arXiv preprint arXiv:1512.08756*, 2015.

[351] F. Ma, R. Chitta, J. Zhou, Q. You, T. Sun, and J. Gao, "Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks," in Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017, pp. 1903-1911.

[352] V. Piccialli, and A. M. Sudoso, "Improving non-intrusive load disaggregation through an attention-based deep neural network," *Energies,* vol. 14, no. 4, pp. 847, 2021.

[353] T. Ha, T. K. Dang, T. T. Dang, T. A. Truong, and M. T. Nguyen, "Differential privacy in deep learning: an overview," in 2019 International Conference on Advanced Computing and Applications (ACOMP), 2019, pp. 97-102.

[354] M. U. Hassan, M. H. Rehmani, and J. Chen, "Differential privacy techniques for cyber physical systems: a survey," *IEEE Communications Surveys & Tutorials*, 2019.

[355] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. Song, "The secret sharer: Evaluating and testing unintended memorization in neural networks," in 28th Security Symposium (Security 19), 2019, pp. 267-284.

[356] N. Agarwal, A. T. Suresh, F. Yu, S. Kumar, and H. B. McMahan, "cpSGD: communication-efficient and differentially-private distributed SGD," in Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 7575-7586.

[357] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," *arXiv preprint arXiv:1712.07557*, 2017.

[358] M. Jawurek, M. Johns, and F. Kerschbaum, "Plug-in privacy for smart metering billing," in International Symposium on Privacy Enhancing Technologies Symposium, 2011, pp. 192-210.

[359] "Raspberry Pi 3 Model B," https://www.raspberrypi.com/products/raspberry-pi-3-model-b/.

[360] O. A. Wahab, A. Mourad, H. Otrok, and T. Taleb, "Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems," *IEEE Communications Surveys & Tutorials,* vol. 23, no. 2, pp. 1342-1397, 2021.

[361] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems,* vol. 32, pp. 8026-8037, 2019.

[362] C. Radebaugh, and U. Erlingsson, "Introducing tensorflow privacy: learning with differential privacy for training data," *Medium. com (accessed 2020-01-27). https://medium. com/tensorflow/introducing-tensorflowprivacy-learning-with-differential-privacy-for-trainingdata-b143c5e801b6*, 2019.

[363] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in Artificial intelligence and statistics, 2017, pp. 1273-1282.

[364] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, 2016, pp. 308-318.

[365] M. Saviozzi, S. Massucco, and F. Silvestro, "Implementation of advanced functionalities for Distribution Management Systems: Load forecasting and modeling through Artificial Neural Networks ensembles," *Electric Power Systems Research,* vol. 167, pp. 230-239, 2019/02/01/, 2019.

[366] J. C. López, M. J. Rider, and Q. Wu, "Parsimonious Short-Term Load Forecasting for Optimal Operation Planning of Electrical Distribution Systems," *IEEE Transactions on Power Systems,* vol. 34, no. 2, pp. 1427-1437, 2019.

[367] P. Singh, G. Pradhan, and S. S, "Denoising of ECG signal by non-local estimation of approximation coefficients in DWT," *Biocybernetics and Biomedical Engineering,* vol. 37, no. 3, pp. 599-610, 2017/01/01/, 2017.

[368] J. Gilles, "Empirical wavelet transform," *IEEE transactions on signal processing,* vol. 61, no. 16, pp. 3999-4010, 2013.

[369] K. Thirumala, A. C. Umarikar, and T. Jain, "Estimation of single-phase and three-phase power-quality indices using empirical wavelet transform," *IEEE Transactions on power delivery,* vol. 30, no. 1, pp. 445-454, 2014.

[370] T. Liu, Z. Luo, J. Huang, and S. Yan, "A comparative study of four kinds of adaptive decomposition algorithms and their applications," *Sensors,* vol. 18, no. 7, pp. 2120, 2018.

[371] M. E. Hodgson, "Searching methods for rapid grid interpolation," *The Professional Geographer,* vol. 41, no. 1, pp. 51-61, 1989.

[372] O. Kramer, D. E. Ciaurri, and S. Koziel, "Derivative-Free Optimization," *Computational Optimization, Methods and Algorithms*, S. Koziel and X.-S. Yang, eds., pp. 61-83, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[373] A. H. Victoria, and G. Maragatham, "Automatic tuning of hyperparameters using Bayesian optimization," *Evolving Systems*, 2020/05/25, 2020.

[374] M. Zhao, and J. Li, "Tuning the hyper-parameters of CMA-ES with tree-structured Parzen estimators," in 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), 2018, pp. 613-618.

[375] I. Couckuyt, D. Deschrijver, and T. Dhaene, "Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization," *Journal of Global Optimization,* vol. 60, no. 3, pp. 575-594, 2014.

[376] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in neural information processing systems,* vol. 25, 2012.

[377] Z. Birnbaum, and R. McCarty, "A Distribution-Free Upper Confidence Bound for \Pr{Y<X\}, Based on Independent Samples of X and Y," *The Annals of Mathematical Statistics*, pp. 558-562, 1958.

[378] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proceedings of the IEEE,* vol. 104, no. 1, pp. 148-175, 2015.

[379] G. R. Lee, R. Gommers, F. Waselewski, K. Wohlfahrt, and A. O'Leary, "PyWavelets: A Python package for wavelet analysis," *Journal of Open Source Software,* vol. 4, no. 36, pp. 1237, 2019.

[380] D. Laszuk, "PyEMD Documentation," 2020.

[381] V. R. Carvalho, M. F. Moraes, A. P. Braga, and E. M. Mendes, "Evaluating five different adaptive decomposition methods for EEG signal seizure detection and classification," *Biomedical Signal Processing Control,* vol. 62, pp. 102073, 2020.

[382] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms," in Proceedings of the 12th Python in science conference, 2013, pp. 20.

[383] X. Y. Zhang, S. Kuenzel, J.-R. Córdoba-Pachón, and C. Watkins, "Privacy-Functionality Trade-Off: A Privacy-Preserving Multi-Channel Smart Metering System," *Energies,* vol. 13, 2020.

[384] N. Blair, A. P. Dobos, J. Freeman, T. Neises, M. Wagner, T. Ferguson, P. Gilman, and S. Janzou, *System advisor model, sam 2014.1. 14: General description*, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2014.

[385] I. S. Data, "National Climatic Data Center (NCDC)," *Asheville, NC*, 2001.

[386] A. Zeyer, R. Schlüter, and H. Ney, “Towards Online-Recognition with Deep Bidirectional LSTM Acoustic Models,” in Interspeech 2016, 2016.

[387] M. Alhussein, K. Aurangzeb, and S. I. Haider, “Hybrid CNN-LSTM model for short-term individual household load forecasting,” *IEEE Access,* vol. 8, pp. 180544-180557, 2020.

[388] T.-Y. Kim, and S.-B. Cho, “Predicting residential energy consumption using CNN-LSTM neural networks,” *Energy,* vol. 182, pp. 72-81, 2019.

[389] J. Xie, J. Fang, C. Liu, and X. Li, “Deep learning-based spectrum sensing in cognitive radio: A CNN-LSTM approach,” *IEEE Communications Letters,* vol. 24, no. 10, pp. 2196-2200, 2020.

[390] L. Meng, E. R. Sanseverino, A. Luna, T. Dragicevic, J. C. Vasquez, and J. M. Guerrero, “Microgrid supervisory controllers and energy management systems: A literature review,” *Renewable and Sustainable Energy Reviews,* vol. 60, pp. 1263-1273, 2016/07/01/, 2016.

[391] S. J. Pan, and Y. Qiang, “A Survey on Transfer Learning,” *IEEE Transactions on Knowledge Data Engineering,* vol. 22, no. 10, pp. 1345-1359, 2010.

[392] A. Arnold, R. Nallapati, and W. W. Cohen, “A Comparative Study of Methods for Transductive Transfer Learning,” in Workshops Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA, 2007.

[393] F. Saghezchi, F. Saghezchi, A. Nascimento, and J. Rodriguez, “Game theory and pricing strategies for demand-side management in the smart grid,” in 2014 9th International Symposium on Communication Systems, Networks & Digital Sign (CSNDSP), 2014, pp. 883-887.

[394] P. Kohlhepp, H. Harb, H. Wolisz, S. Waczowicz, D. Müller, and V. Hagenmeyer, “Large-scale grid integration of residential thermal energy storages as demand-side flexibility resource: A review of international field studies,” *Renewable and Sustainable Energy Reviews,* vol. 101, pp. 527-547, 2019/03/01/, 2019.

[395] Y. M. Lee, R. Horesh, and L. Liberti, “Optimal HVAC Control as Demand Response with On-site Energy Storage and Generation System,” *Energy Procedia,* vol. 78, pp. 2106-2111, 2015/11/01/, 2015.

[396] C. Xue, H. Zhou, Q. Wu, X. Wu, and X. Xu, “Impact of Incentive Policies and Other Socio-Economic Factors on Electric Vehicle Market Share: A Panel Data Analysis from the 20 Countries,” *Sustainability,* vol. 13, no. 5, pp. 2928, 2021.

[397] C. Weare, *The California electricity crisis: causes and policy options*: Public Policy Instit. of CA, 2003.

[398] R. Cox, "Heating, Ventilating and Air Conditioning Systems," *Energy Management Handbook*, pp. 261-297: River Publishers, 2020.

[399] t. O. W. o. T. C. o. Austin. "Official City Holidays," https://www.austintexas.gov/department/official-city-holidays.

[400] T. Feddern-Bekcan, "Google calendar," *Journal of the Medical Library Association: JMLA,* vol. 96, no. 4, pp. 394, 2008.

[401] R. Koenker, and G. Bassett Jr, "Regression quantiles," *Econometrica: journal of the Econometric Society*, pp. 33-50, 1978.

[402] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, "Scikit-learn: Machine learning in Python," *the Journal of machine Learning research,* vol. 12, pp. 2825-2830, 2011.

[403] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard, "{TensorFlow}: A System for {Large-Scale} Machine Learning," in 12th USENIX symposium on operating systems design and implementation (OSDI 16), 2016, pp. 265-283.

[404] D. S. Wilks, *Statistical methods in the atmospheric sciences*: Academic press, 2011.

[405] R. L. Winkler, "A decision-theoretic approach to interval estimation," *Journal of the American Statistical Association,* vol. 67, no. 337, pp. 187-191, 1972.

[406] H. Hersbach, "Decomposition of the continuous ranked probability score for ensemble prediction systems," *Weather and Forecasting,* vol. 15, no. 5, pp. 559-570, 2000.

[407] G. Petneházi, "QCNN: Quantile Convolutional Neural Network," *arXiv preprint arXiv:1908.07978*, 2019.

[408] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE,* vol. 109, no. 1, pp. 43-76, 2020.

[409] N. Qi, L. Cheng, H. Xu, K. Wu, X. Li, Y. Wang, and R. Liu, "Smart meter data-driven evaluation of operational demand response potential of residential air conditioning loads," *Applied Energy,* vol. 279, pp. 115708, 2020.

[410] J. R. Schofield, R. Carmichael, S. Tindemans, M. Bilton, M. Woolf, and G. Strbac, "Low Carbon London project: Data from the dynamic time-of-use electricity pricing trial, 2013," *uK Data Service, SN,* vol. 7857, no. 2015, pp. 7857-1, 2015.

Appendix A Variational Mode Decomposition and Empirical Mode Decomposition

In Appendix A, two mode decomposition techniques, variational mode decomposition (VMD), and empirical mode decomposition (EMD), are introduced. These two methods are employed as benchmarks in Chapter 6.

A.1 Empirical Mode Decomposition

EMD is a self-adoptive mode decomposition method, it can decompose the signal in temporal space directly without transferring the signal into frequency space. The characteristic of EMD is it does not reply on any mathematical functions but adopt to the signal $f(t)$ accordingly, $f(t)$ is decomposed into $N+1$ Intrinsic Mode Functions (IMFs) $f_k(t)$ and a residuum $r(t)$, see (A-1).

$$f(t) = \sum_{k=0}^{N} f_k(t) + r(t) \tag{A-1}$$

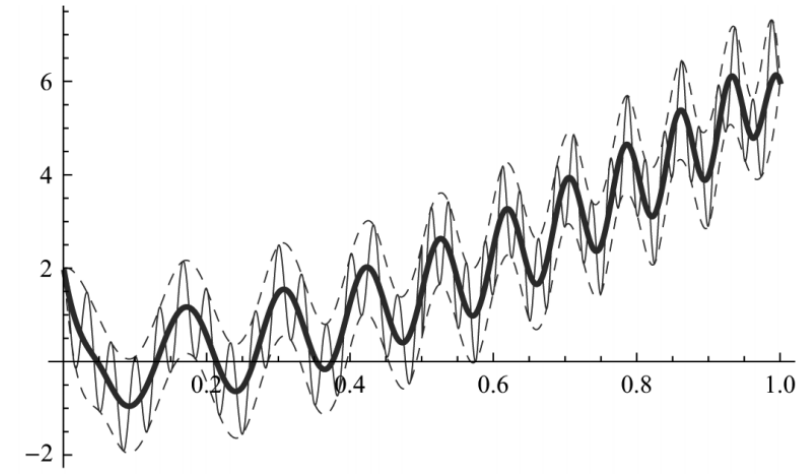
An IMF is an AM-FM function and can be expressed as follow:

$$f_k(t) = F_k(t)\cos\big(\varphi_k(t)\big) \quad \text{where } F_k(t), \varphi_k'(t) > 0 \quad \forall t \tag{A-2}$$

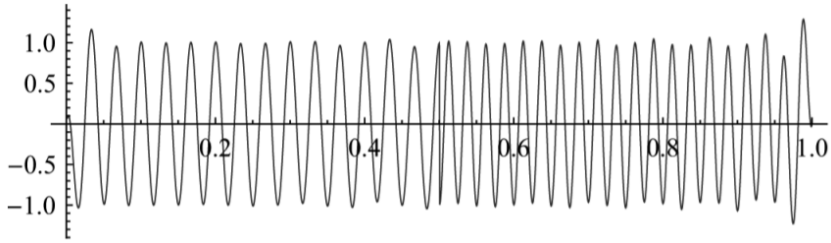
The main assumption is that $F_k(t)$ and $\varphi_k'(t)$ varies much slower than $\varphi_k(t)$. An The detailed process is presented in Program A-1. An IMF should satisfy two conditions: (1) the number of extrema and the number of zero crossings must be the same or differ at most by one; 2) at any point, the mean value of upper (defined by local maxima) and lower envelope (defined by local minima) is zero. The detailed process of EMD is demonstrated in Algorithm A-1.

Algorithm **A-1**: Empirical Mode Decomposition (EMD).

Input: Real-world signal $f(t)$.
Output: IMFs f_k, where $k = 1,2, \dots, N+1$.
Initialization: $n := 1$, $r_0(t) = f(t)$.
Step 1: Extract the nth IMF as follows:
a): Initialize $h_0(t) := r_{n-1}(t)$ and $k := 1$.
b): Detect the maxima and minima of $h_{k-1}(t)$.
c): Compute the upper and lower envelope, $U_{k-1}(t)$ and $L_{k-1}(t)$ by a cubic spline interpolation from the maxima and minima (See Figure B-1 (a)).
d): Compute the mean envelope: $m_{k-1}(t) = \frac{U_{k-1}(t)+L_{k-1}(t)}{2}$.
e): Obtain the candidate component: $h_k(t) := h_{k-1}(t) - m_{k-1}(t)$ (See Figure B-1 (b)).
f): If $r_1(t)$ satisfies conditions of an IMF:
i): $x_n(t) := h_k(t)$ and $r_n(t) := r_{n-1}(t) - x_n(t)$.
g): Else:
i): $k := k + 1$.
ii): Repeat step b)-g) until $h_k(t)$ is an IMF.
Step 2: If $r_n(t)$ is a residuum, stop the process.
Else $n := n + 1$ and start from Step 1.



(a)



(b)

Figure A-1. (a) EMD: basic IMF detection; (b) the : the first IMF candidate.

A.2 Variational Mode Decomposition

Variational mode decomposition (VMD) is proposed by K. Ragomiretskiy in 2014, it is a non-recursive, adaptive decomposition estimation method to decompose the original signal into K mode functions $u_k(t)$ with specific bandwidth in the frequency

domain. And each $u_k(t)$ is concentrated near the central frequency ω_k. The nature of VMD method is an optimal process to look for K modes that make the overall bandwidth smallest, shown in (A-3):

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_{k=1}^{K} \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|^2 \right\} \text{ s.t. } \sum_{k=1}^{K} u_k = f(t) \quad \text{(A-3)}$$

where $f(t)$ is the original signal, $\delta(t)$ represents Dirac distribution function; $\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t)$ is the corresponding unilateral spectrum of $u_k(t)$ by implementing Hilbert transformation; u_k and ω_k represents the k th mode and corresponding central frequency; $e^{-j\omega_k t}$ is the exponent term to adjust the frequency spectrum to the corresponding base frequency band.

Then by introducing a quadratic penalty α and Lagrange multiplier operator $\lambda(t)$, the constrained problem mentioned in (A-3) is transformed into a non-constrained problem, the augmented Lagrangian expression is expressed as:

$$\begin{aligned} L(\{u_k\},\{\omega_k\},\lambda) = \alpha \sum_{k=1}^{K} & \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \\ & + \left\| f(t) - \sum_{k=1}^{K} u_k(t) \right\|_2^2 \\ & + \left\langle \lambda(t), f(t) - \sum_{k=1}^{K} u_k(t) \right\rangle \end{aligned} \quad \text{(A-4)}$$

where α is adopted to ensure the accuracy of the reconstruction; and $\lambda(t)$ is employed to tighten the constraint; and $\left\| f(t) - \sum_{k=1}^{K} u_k(t) \right\|_2^2$ is a quadratic penalty term to speed up the convergence. The expression (A-4) can be solved by employing alternate direction method of multipliers (ADMM) to compute the saddle point of the equation. According to ADMM optimization method, u_k and ω_k is updated as:

$$u_k^{n+1} = \arg \min_{u_k} L(\{u_{i<k}^{n+1}\},\{u_{i\geq k}^{n}\},\{\omega_i^n\},\{\lambda^n\}) \quad \text{(A-5)}$$

$$\omega_k^{n+1} = \arg \min_{\omega_k} L(\{u_i^{n+1}\},\{\omega_{i<k}^{n+1}\},\{\omega_{i\geq k}^{n}\},\{\lambda^n\}) \quad \text{(A-6)}$$

$$\lambda^{n+1} = \lambda^n + \tau(f(t) - \sum_{k=1}^{K} u_k^{n+1}) \quad \text{(A-7)}$$

$$\sum_{k=1}^{K} \left\| u_k^{n+1} - u_k^n \right\|_2^2 / \left\| u_k^n \right\|_2^2 < \varepsilon \quad \text{(A-8)}$$

Finally, u_k^{n+1} and ω_k^{n+1} are solved as:

$$u_k^{n+1}(\omega) = \frac{f(\omega) - \sum_{i \neq k} u_i(\omega) + \frac{\lambda(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2} \tag{A-9}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega \left| u_k^{n+1}(\omega) \right|^2 d\omega}{\int_0^\infty \left| u_k^{n+1}(\omega) \right|^2 d\omega} \tag{A-10}$$

where $f(\omega)$, $\lambda(\omega)$, $u_i(\omega)$, $u_k^{n+1}(\omega)$ represent the Fourier transform of $f(t)$, $\lambda(t)$, $u_i(t)$, $u_k^{n+1}(t)$.